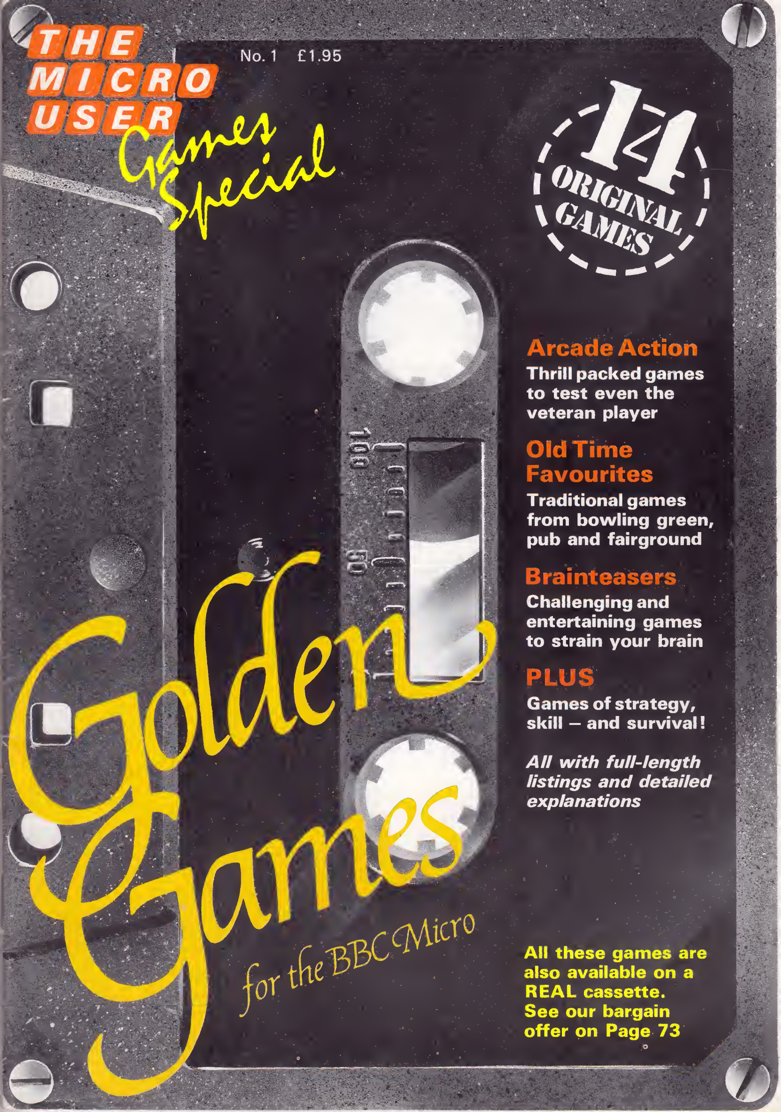
THE MICRO USER

Games Special

No. 1 £1.95



14 ORIGINAL GAMES

Golden Games for the BBC Micro

Arcade Action

Thrill packed games to test even the veteran player

Old Time Favourites

Traditional games from bowling green, pub and fairground

Brainteasers

Challenging and entertaining games to strain your brain

PLUS

Games of strategy, skill – and survival!

All with full-length listings and detailed explanations

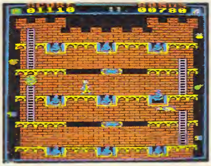
All these games are also available on a REAL cassette. See our bargain offer on Page 73

GAMES

FOR THE BBC 32K O.S. 1.2

Q*MAN

CASTLE ASSAULT

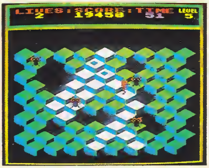


Q*MAN'S BROTHER

NIGHTMARE MAZE

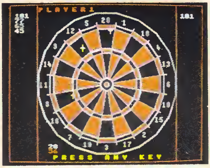
(Also available for the Electron)

SCREWBALL



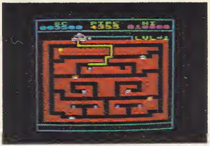
CASTLE OF GEMS

DARTS



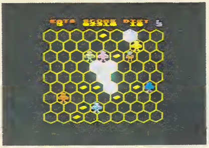
3-D MUNCHY

DIAMOND MINE



(ALSO AVAILABLE FOR ELECTRON)

GUY IN THE HAT



ARTIST £6.95

BANANA MAN

MRM SOFTWARE

ALL GAMES ONLY

£5.70 FULLY INCLUSIVE

Boots

17 Cross Coates Road, Grimsby, South Humberside. Tel: 0472 44304.

THE MICRO USER
Games Special

Managing Editor
Derek Meakin

Features Editor
Mike Bibby

Production Editor
Peter Glover

Layout Design
Heather Sheldrick

Associate Editor
Alan McLachlan

Advertisement Manager
John Riding

Advertising Sales
John Snowden

Editor in Chief,
Database Publications
Peter Brameld

Tel: 061-456 8383 (Editorial)
061-456 8500 (Advertising)
061-480 0171 (Subscriptions)
Telex: 667664 SHARET G
Prestel Mailbox: 614568383

Published by:
Database Publications Ltd,
Europa House, 68 Chester Road,
Hazel Grove, Stockport SK7 5NY.

ABC Member of Audit Bureau of Circulations

The Micro User welcomes program listings and articles for publication. Material should be typed or computer-printed, and preferably double-spaced. Program listings should be accompanied by cassette tape or disc. Please enclose a stamped, self-addressed envelope, otherwise the return of material cannot be guaranteed. Contributions accepted for publication will be on an all-rights basis.

The Micro User is an independent publication and neither the BBC nor Acorn Computers Ltd are responsible for any of the articles in this issue or for any of the opinions expressed.

News trade distribution:
Europress Sales and Distribution Limited, 11 Brighton Road, Crawley, West Sussex RH10 6AF. Circulation 0293 27053.

CONTENTS

NOW GET OUT

You've got monster-size problems in Explorer Eddie – a nerve-wracking split level game created by BARRY WAKELIN

HAVE you got problems! Trapped in the darkest recesses of a monster-infested mine, you have to battle your way upwards through 12 levels.

There is a lift to help, but you'll have to collect all the keys on each level before you can use it safely.

You can walk on and jump off the ledges and conveyor belts found on all levels, but the bricks cover the cavern floor completely. Be careful though – if you jump too far you're dead!

You'll have to watch the conveyor belts as well – you can't walk against the direction of travel. You'll stand still if you try!

And, of course, the final level has its own special trick, but you wouldn't want us to spoil it by telling, would you?

You can even tailor the game to your own design.

You'll have to change the user-defined characters 239 to 254 at the correct place in the program for the level being altered, then redefine logical colours 2 and 1.

Next, give the start position and direction of movement for the two hazards.

You must make sure that if the hazard moves horizontally there is some sort of ground beneath it and that there is a gap in the ground where it is to stop.

If the hazard moves vertically there must be pieces of ground where the hazard is to stop.

Next, give the name of this level. Then set the number of pieces of ground. Print, at their appropriate positions, the keys(K$), stalactites(S$) and trees (T$).

Now go to the line where the positions of that level's pieces of ground are stored and enter your own data. Each piece of ground is stored as X position, Y position, length and type of ground.

Type 1 is a ledge, type 2 is brick, type 3 is a left conveyor and type 4 is a right conveyor.

There are 24 different monsters, each of 2 × 2 character size. All this takes up rather a lot of memory, so take care not to add unnecessary spaces and so on when typing it in.

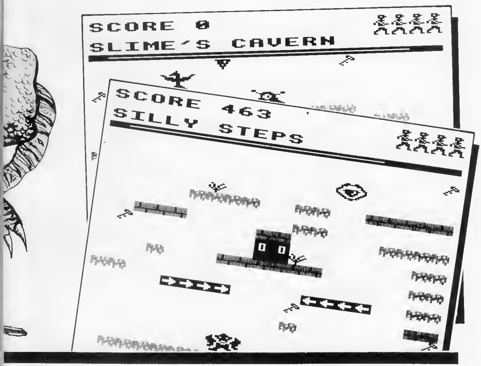
KEYS

Z	*left*
X	*right*
Return	*jump*

VARIABLES

P$(n)	Character used for the player.
M$(n)	First character for each monster.
N$(n)	Second character for the monsters.
D%(n)	Horizontal direction of monster n.
E%(n)	Vertical direction of monster n.
A%(n)	X position of monster n.
B%(n)	Y position of monster n.
HI$(n)	Name of high scorer n.
HI%(n)	Score of high scorer n.
I%,I	General variables.
K$	Key character.
S$	Stalactite character.
T$	Tree character.
G$	Puts CGOL3,3 in the P$(n) characters.
C$	Puts cursor down and left in characters.
L%	Character number being used for monster 0.
LEV%	Level being played.
LIV%	Number of lives.
SCR%	Score.
FI%	True if level 13 is reached.
W%	Number of keys collected.
O%	Last character printed for the player's character.
C%	Character being used for the player's character.
J%	True if you are jumping.
V%	Vertical movement of player.
DE%	True if hit by monster.
TIM%	Time remaining.
D%	Used to store direction of motion before a jump.
X%	X position of player.
Y%	Y position of player.
Q%	Number of pieces of ground used on this level.
S%	Start of piece of ground.
F%	Length of piece of ground.

OF THAT!



K%	Type of ground.
T%	X position of lift.
U%	Y position of lift.
K%	When elsewhere in the program it is used as the character for monster 1.
H%	Horizontal move of player.
Z%	Stage of jump.
A%	This is two if the player is going left.
R%	Ranking on table.
W$	String to be printed in double height.
N$	Name of level.

DESCRIPTION OF LINES

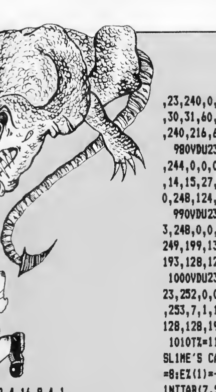
10-80	Set up characters, arrays, variables and envelopes.
90-110	Set up screen.
120-150	Move the monsters if they move vertically.
160-170	Move the monsters if they move horizontally.
180-190	Print monsters at their new positions.
200	End if player has been hit.
210	Decrement clock.
220	Check if the player is jumping.
230-240	Check for left, right input.
250-260	Check if player is on a conveyor belt.
270	Check for jump input.
280-330	Act on move.
340-370	Move player and restart loop.
380-480	Checks and moves if jumping.
490-500	Subroutine if a key is grabbed.
510-520	Lose a life.
530-540	End of game.
550-580	Input name if on score board.
590-620	Print up high score table.

```
   5REM  (C) The Micro User
  10MODE5:DIMP$(3),M$(1),N$(1),D%(1)
,E%(1),A%(1),B%(1),HI$(8),HI%(8):FORI
%=1TO8:HI$(I%)="The Micro User":HI%(I
%)=100:NEXT
  20ENVELOPE1,1,5,0,0,25,0,0,126,0,0
,-126,126,126:ENVELOPE2,1,0,0,0,0,0,0
,127,-3,-2,-1,126,0
  30VDU23,224,28,22,28,8,28,40,84,91
,23,225,28,0,28,30,54,98,66,99,23,226
,92,0,28,28,20,20,20,22,23,227,56,104
,56,16,56,20,42,218,23,228,56,0,56,12
0,108,70,66,198,23,229,56,104,56,16,5
6,16,104,180,23,255,12,10,5,11,16,40,
64,32
  40VDU23,233,28,22,28,8,28,8,22,45,
23,234,58,0,56,56,40,40,40,104,23,235
,0,255,169,171,137,171,169,255
  50VDU23,230,66,36,169,170,84,56,16
,16,23,231,255,239,207,129,129,207,23
9,255,23,232,255,247,243,129,129,243,
247,255,23,236,206,251,239,255,189,24
7,166,34,23,237,223,223,223,223,0,251
,251,251,23,238,127,59,58,62,20,28,8,
8
  60K$=CHR$255:S$=CHR$238:T$=CHR$230
:G$=CHR$18+CHR$3+CHR$3:C$=CHR$10+CHR$
8:P$(0)=G$+CHR$224+C$+CHR$225:P$(1)=G
$+CHR$233+C$+CHR$226:P$(2)=G$+CHR$227
+C$+CHR$228:P$(3)=G$+CHR$229+C$+CHR$2
34:C$=C$+CHR$8:R$="  "+C$+"  ":GOTO59
0
  70LEV%=1:LIV%=5:SCR%=0
  80G$=CHR$17+CHR$2:M$(0)=G$+CHR$239
+CHR$240+C$+CHR$241+CHR$242:M$(1)=G$+
CHR$243+CHR$244+C$+CHR$245+CHR$246:L%
=0:N$(0)=G$+CHR$247+CHR$248+C$+CHR$24
9+CHR$250:N$(1)=G$+CHR$251+CHR$252+C$
+CHR$253+CHR$254
  90RESTORE(LEV%*20+640):COLOUR1:PRI
NTTAB(0,30)STRING$(20,CHR$236);:FI%=0
:D%(0)=0:D%(1)=0:E%(0)=0:E%(1)=0:COLO
UR3:GOSUB(LEV%*60+910):IFFI%=1GOTO530
 100W%=0:O%=0:C%=0:A%=0:J%=0:V%=0:DE
%=0:TIM%=1200:D%=0:X%=0:Y%=124:FORI%=
1TOQ%:READS%,P%,F%,K%:GOSUB(K%*10+900
):NEXT:PROClft(T%,U%):K%=0:GCOL3,3:VD
U5:MOVEX%,Y%:PRINTTAB(X%,Y%)P$(0):VDU
4
 110COLOUR3:COLOUR128:PRINTTAB(0,1)"
SCORE ";SCR%TAB(16,1)STRING$(LIV%-1,P
$(0)+CHR$11)TAB(0,3)N$:VDU23,1;0;0;0;
0;
 120K%=(K%+1)MOD2:L%=(L%+1)MOD2:FORI
%=0TO1:IFD%(I%)<>0 GOTO160
 130IFE%(I%)=-1ANDPOINT(A%(I%)*64,(3
2-B%(I%))*32+16)>0 E%(I%)=1
 140IFE%(I%)=1ANDPOINT(A%(I%)*64,(32
-B%(I%))*32-80)>0 E%(I%)=-1
```

→

From Page 5

```
  150GOTO180
  160IFD%(I%)=-1ANDPOINT(A%(I%)*64-64
,(32-B%(I%))*32-76)=0D%(I%)=1
  170IFD%(I%)=1ANDPOINT(A%(I%)*64+128
,(32-B%(I%))*32-76)=0D%(I%)=-1
  180A%(I%)=A%(I%)+D%(I%):B%(I%)=B%(I
%)+E%(I%):IFA%(I%)*64>=X%-64ANDA%(I%)
*64<X%+64AND(32-B%(I%))*32>Y%-36AND(3
2-B%(I%))*32<Y%+68DE%=1:VDU5:GCOL3,3:
MOVEX%,Y%:PRINTP$(O%):Y%=Y%-92:MOVEX%
,Y%:PRINTP$(O%):VDU4
  190NEXT:PRINTTAB(A%(0)-D%(0),B%(0)-
E%(0))R$TAB(A%(0),B%(0))M$(K%)TAB(A%(
1)-D%(1),B%(1)-E%(1))R$TAB(A%(1),B%(1
))N$(L%)
  200IFDE%=1GOTO510
  210GCOL0,3:PLOT69,TIM%,876:TIM%=TIM
%-4:IFTIM%<756GOTO510
  220IFJ%=1GOTO380
  230H%=0:IFINKEY(-98)H%=-64
  240IFINKEY(-67)H%=64
  250IFPOINT(X%+H%+32,Y%-84)=2H%=H%-6
4:IFH%<-64H%=-64
  260IFPOINT(X%+H%+24,Y%-84)=2H%=H%+6
4:IFH%>64H%=64
  270IFINKEY(-74)J%=1:Z%=0:D%=H%:H%=0
:GOTO380
  280IFX%+H%<0ORX%+H%>1216H%=0
  290IFPOINT(X%+H%,Y%-64)=0J%=1:Z%=4:
D%=0
  300IFLEV%=12ANDPOINT(X%+H%,Y%-76)=3
GOSUB630
  310IFH%=0ANDV%=0FORI=1TO150:NEXT:GO
TO120
  320IF(X%+H%>=T%ANDX%+H%<T%+128ANDY%
+V%=U%)ANDW%>3GOTO640
  330IFPOINT(X%+H%+8,Y%+V%-32)=3GOTO5
10
  340VDU5:A%=0:IFH%=-64ORD%=-64A%=2
  350C%=(C%+1)MOD2:X%=X%+H%:Y%=Y%+V%:
MOVEX%-H%,Y%-V%:PRINTP$(O%):MOVEX%,Y%
:PRINTP$(C%+A%):O%=C%+A%:VDU4,23,1;0;
0;0;0;
  360IF(A%(0)*64>=X%-64ANDA%(0)*64<X%
+64AND(32-B%(0))*32>Y%-36AND(32-B%(0)
)*32<Y%+68)OR(A%(1)*64>=X%-64ANDA%(1)
*64<X%+64AND(32-B%(1))*32>Y%-36AND(32
-B%(1))*32<Y%+68)GOTO510
  370GOTO120
  380Z%=Z%+1:SOUND1,-10,Z%*4,1:H%=0:I
FZ%>2ANDZ%<6H%=D%
  390IFX%+H%<0ORX%+H%>1216H%=0
  400V%=32:IFZ%>3V%=-32
  410IFY%+V%<92ORY%+V%>860V%=0
  420IF(V%=32ANDPOINT(X%+H%+8,Y%+V%)=
3)OR(V%=-32ANDPOINT(X%+H%+8,Y%-64)=3)
GOTO510
  430IFV%=-32ANDPOINT(X%+H%,Y%-64)>0J
%=0:V%=0:D%=0:GOTO230
  440IFV%=32ANDPOINT(X%+H%+32,Y%+V%)=
3GOSUB490:MOVEX%+H%,Y%+V%:VDU255
  450IFV%=32ANDPOINT(X%+H%+8,Y%+16)=2
V%=0:Z%=3
  460IFV%=-32ANDPOINT(X%+H%+32,Y%-64)
=3GOSUB490:MOVEX%+H%,Y%-64:VDU255
  470IFZ%=1GOTO510
  480GOTO310
  490SOUND2,2,150,1:W%=W%+1:SCR%=SCR%
+LEV%*25:COLOUR3:PRINTTAB(6,1);SCR%:V
DU5:GCOL0,0:TIM%=TIM%+64:IFTIM%>1264T
IM%=1264
  500RETURN
  510VDU5:GCOL3,3
  520REPEAT:Y%=Y%-32:MOVEX%,Y%+32:PRI
NTP$(O%):MOVEX%,Y%:PRINTP$(O%):SOUND1
,-12,Y%/4,2:FORI=1TO150:NEXT:UNTILY%<
128:SOUND0,2,6,1:LIV%=LIV%-1:IFLIV%>0
MODE5:GOTO80
  530FORI=1TO3000:NEXT:MODE7:IFSCR%<H
I%(8)GOTO590
  540*FX15,1
  550PRINTTAB(8,3)CHR$136"THAT'S A NE
W HI-SCORE"TAB(10,7)CHR$130"What is y
our name"
  560INPUTTAB(0,9)P$:IFLEN(P$)>24PRIN
TTAB(15,12)CHR$129"Too long.":FORI=1T
O5000:NEXT:CLS:GOTO550
  570R%=9:REPEAT:R%=R%-1:UNTILR%=1ORH
I%(R%-1)>SCR%
  580FORI%=8TOR%+1STEP-1:HI$(I%)=HI$(
I%-1):HI%(I%)=HI%(I%-1):NEXT:HI$(R%)=
P$:HI%(R%)=SCR%
  590MODE7:FORI%=0TO1:PRINTTAB(4,I%)C
HR$141CHR$129"EXPLORER EDDIES HALL OF
 FAME":NEXT:FORI%=1TO8:PRINTTAB(0,I%*
2+3);I%TAB(3,I%*2+3)STRING$(28,".")TA
B(5,I%*2+3)HI$(I%)TAB(33,I%*2+3);HI%(
I%):NEXT
  600*FX15,1
  610PRINTTAB(1,3)CHR$130"Collect the
 keys to escape the mine."TAB(2,21)CH
R$133"Z - Left  X - Right  RETURN - J
ump"TAB(5,23)CHR$131"Press any key to
 play again."
  620Q%=GET:MODE5:GOTO70
  630D%(0)=0:D%(1)=0:E%(0)=0:E%(1)=0:
SOUND0,2,6,1:VDU19,0,3;0;:FORI=1TO100
:NEXT:VDU19,0,0;0;5:GCOL0,0:MOVEX%+H%
,Y%-64:VDU235,4:RETURN
  640GCOL0,3:COLOUR3:VDU4:FORI%=TIM%T
O75STEP-8:PLOT69,I%,876:SCR%=SCR%+5:P
RINTTAB(6,1);SCR%:SOUND1,-13,75,.1:NE
XT:LEV%=LEV%+1:FORI=1TO3000:NEXT:MODE
5:GOTO80
  650DEFPROCdbl(W$,Y%):W$=CHR$141+W$:
T%=(40-LEN(W$))/2-1:PRINTTAB(T%,Y%)CH
R$130W$TAB(T%,Y%+1)CHR$134W$:ENDPROC
  660DATA0,26,8,1,9,27,11,1,5,23,3,3,
14,21,6,2,7,20,3,1,0,19,3,1,10,18,6,1
,4,17,2,2
  670DATA16,17,2,2,0,14,4,1,16,15,2,1
,16,13,2,1,16,11,2,1,2,11,2,1,6,10,9,
1
  680DATA0,27,2,1,3,25,4,1,9,25,4,2,1
5,25,3,1,18,22,2,2,3,20,4,1,9,20,4,3,
15,20,3,1
  690DATA0,17,2,1,2,15,2,1,6,15,7,2,1
5,15,5,1,11,12,2,1,15,12,3,2,5,10,2,1
,9,10,2,1,0,7,4,2,11,8,9,1
  700DATA15,28,2,1,1,26,13,1,18,26,2,
1,16,24,2,2,18,21,2,2,0,18,2,1,4,19,4
,4,12,19,4,3,18,19,2,1
  710DATA18,17,2,1,3,16,1,1,7,16,6,2,
18,15,2,1,9,13,2,2,16,13,4,1,2,12,3,2
,11,12,2,1,5,10,4,1,11,10,2,1,15,10,5
,2,11,22,1,1
  720DATA8,27,1,2,5,26,1,1,1,25,2,1,1
1,25,1,2,0,22,1,1,8,23,1,1,13,23,3,1,
11,21,1,1,16,21,4,2,9,18,1,1
  730DATA0,17,4,1,12,16,1,1,15,16,4,1
,7,15,1,1,19,15,1,2,0,14,1,1,4,13,2,1
,0,11,1,1,6,10,13,4,4,9,1,1,0,8,1,1,1
,8,3,4
  740DATA13,27,4,1,15,24,1,1,1,23,4,1
,6,23,1,1,9,23,1,1,12,23,1,1,0,22,1,2
,0,19,4,1,1,17,8,4,11,17,8,4
  750DATA0,16,1,1,9,14,2,2,19,14,1,2,
0,13,1,1,3,12,2,1,16,12,2,3,4,10,1,1,
15,10,1,1,0,8,2,1,5,8,3,1,9,8,5,1,16,
7,4,2
  760DATA0,27,1,1,4,25,13,2,18,27,1,1
,1,24,2,1,3,22,14,2,9,22,2,4,17,21,2,
3,5,18,9,1
  770DATA16,18,2,1,4,16,1,1,1,15,1,1,
3,12,2,4,6,12,3,1,11,12,3,4,17,14,3,2
,13,9,1,1,13,7,5,1,18,9,1,2
  780DATA2,27,1,1,5,26,1,1,15,26,1,1,
9,25,3,4,16,24,1,1,3,23,2,1,12,23,2,1
,17,22,1,1,13,21,2,1,6,20,2,1,18,20,1
,1,9,19,1,1,2,17,1,1,5,17,1,2,9,17,1,
1,12,17,2,4,15,17,1,1,16,17,2,4,19,15
,1,1,2,14,1,1,16,14,1,2,5,13,1,1,5,11
,1,1,6,11,10,4
  790DATA0,8,2,2,2,8,3,3
  800DATA0,27,1,1,1,27,17,4,11,27,2,2
,13,23,1,3,2,21,7,3,10,21,1,1,0,19,2,
1,2,16,1,1,5,16,2,4,11,16,1,2,10,14,1
,1
  810DATA2,13,1,1,7,13,1,2,3,10,1,1,0
,8,2,1,4,7,12,3,16,10,4,1,16,13,1,1,1
6,16,1,1,18,19,1,1,18,22,1,1,18,24,1,
2
  820DATA1,28,1,1,0,25,1,1,3,25,1,1,1
3,25,1,3,16,25,1,3,10,24,1,1,6,23,1,1
,17,23,1,1,2,21,1,2,6,21,1,1,8,20,8,4
,18,20,1,1
  830DATA3,17,7,3,13,17,1,1,16,15,1,1
,19,15,1,1,7,14,1,1,3,13,1,1,19,12,1,
```



```
1,4,11,4,3,1,9,2,4,16,9,4,1
  840DATA8,27,2,2,11,25,1,1,14,25,5,4
,3,23,1,1,6,23,3,4,2,22,1,1,19,22,1,2
,14,20,5,3,3,19,10,3,3,16,7,2,11,16,2
,2,1,14,1,1,9,13,1,4,11,13,1,1,14,13,
2,4,1,12,2,4,5,12,1,1,18,12,1,1,8,10,
1,1,5,9,1,1,15,9,1,1,18,9,1,1,0,8,1,1
  850DATA2,8,1,3,12,8,1,1,9,7,2,1
  860DATA9,27,1,2,11,27,1,2,9,24,1,1,
11,24,1,1,0,22,1,2,1,21,18,4,19,19,1,
1,1,17,18,3,1
  870DATA13,8,3,10,13,9,4,0,10,3,2,16
,10,2,3,13,9,1,1,10,8,1,3,8,8,1,3,6,8
,1,3,5,8,1,2,4,8,1,4,10,16,1,2
  880DATA7,27,1,2,10,26,1,1,16,26,1,1
,8,23,1,3,12,23,4,4,17,23,2,3,1,22,5,
3,1,19,2,2,4,19,2,2,9,19,1,1,12,19,1,
3,13,18,4,4
  890DATA19,17,1,2,1,16,1,1,3,15,1,1,
6,15,6,4,14,18,2,3,1,14,2,4,15,14,1,3
,18,14,2,3,0,10,3,2,5,11,10,2,3,10,1,
4
  900DATA0
  910COLOUR1:COLOUR128:PRINTTAB(S%,P%
)STRING$(F%,CHR$236);:RETURN
  920COLOUR1:COLOUR130:PRINTTAB(S%,P%
)STRING$(F%,CHR$237);:RETURN
  930COLOUR2:COLOUR128:PRINTTAB(S%,P%
)STRING$(F%,CHR$231);:RETURN
  940COLOUR2:COLOUR128:PRINTTAB(S%,P%
)STRING$(F%,CHR$232);:RETURN
  950DEFPROClft(T%,U%):GCOL0,3:MOVET%
,U%:MOVET%+120,U%:PLOT85,T%+120,U%-64
:MOVET%,U%-64:PLOT85,T%,U%:GCOL0,0:MO
VET%+16,U%-12:DRAWT%+32,U%-12:DRAWT%+
32,U%-32:DRAWT%+16,U%-32:DRAWT%+16,U%
-12:MOVET%+104,U%-12:DRAWT%+104,U%-32
  960DRAWT%+92,U%-32:DRAWT%+92,U%-12:
DRAWT%+104,U%-12:GCOL0,3:FORI%=868TO8
84STEP4:MOVE0,I%:DRAW1279,I%:NEXT:GCO
L0,0:MOVE79,876:DRAW1200,876:ENDPROC
  970VDU23,239,0,0,0,195,199,62,13,13
,23,240,0,6,6,200,240,96,32,32,23,241
,30,31,60,120,251,63,30,12,23,242,112
,240,216,60,126,255,239,70
  980VDU23,243,0,96,96,19,15,6,4,4,23
,244,0,0,0,195,227,124,176,176,23,245
,14,15,27,56,124,255,247,98,23,246,12
0,248,124,62,223,252,120,48
  990VDU23,247,1,3,5,15,1,25,63,127,2
3,248,0,0,128,128,128,152,252,254,23,
249,199,131,1,1,3,7,12,16,23,250,227,
193,128,128,192,224,48,8
 1000VDU23,251,1,3,7,207,97,61,63,31,
23,252,0,0,128,131,134,188,252,248,23
,253,7,1,1,1,3,7,12,8,23,254,224,128,
128,128,192,224,48,16
 1010T%=1152:U%=124:VDU19,2,2;0;:N$="
SLIME'S CAVERN":D%(0)=1:A%(0)=6:B%(0)
=8:E%(1)=-1:A%(1)=4:B%(1)=15:Q%=15:PR
INTTAB(7,5)S$TAB(5,18)S$TAB(0,8)K$TAB
(14,5)K$TAB(13,14)K$TAB(18,15)K$
 1020RETURN
 1030VDU23,239,3,14,61,123,247,206,4,
0,23,240,192,112,188,222,239,115,32,1
28,23,241,0,0,0,0,0,0,0,0,23,242,128,
128,128,128,128,144,80,32
 1040VDU23,243,1,2,5,13,11,11,27,21,2
3,244,128,64,160,176,208,208,216,168,
23,245,25,8,0,0,0,4,5,2,23,246,152,16
,128,128,128,128,0,0
 1050VDU23,247,7,31,60,56,116,226,193
,194,23,248,224,248,60,30,46,71,131,6
7,23,249,194,193,226,116,56,60,31,7,2
3,250,67,131,71,46,30,60,248,224
 1060VDU23,251,7,31,60,48,112,224,193
,194,23,252,224,248,188,140,142,135,1
31,127,23,253,254,193,225,113,49,61,3
1,7,23,254,67,131,7,14,12,60,248,224
 1070VDU19,2,6;0;19,1,4;0;:A%(0)=7:B%
(0)=16:E%(0)=1:A%(1)=13:B%(1)=28:E%(1
)=-1:N$="LOOPY LEVELS":Q%=18:PRINTTAB
(19,18)K$TAB(0,13)K$TAB(18,12)K$TAB(1
7,5)K$TAB(4,19)T$TAB(15,7)T$:T%=0:U%=
860
 1080RETURN
 1090VDU23,239,0,6,15,28,16,48,103,79
,23,240,0,96,240,56,8,12,230,178,23,2
41,71,103,51,25,12,7,1,0,23,242,162,1
02,204,152,48,224,128,0
 1100VDU23,243,0,0,0,0,6,15,31,51,23,
244,0,0,0,0,96,240,248,204,23,245,28,
15,7,1,0,0,0,0,23,246,56,248,240,128,
0,0,0,0
 1110VDU23,247,1,3,47,109,231,243,250
,121,23,248,128,192,244,182,231,207,9
5,158,23,249,118,55,119,103,14,60,60,
0,23,250,110,236,238,230,112,48,60,60
 1120VDU23,251,1,3,47,109,231,243,251
,248,23,252,128,192,244,182,231,207,2
23,31,23,253,119,55,119,103,14,12,60,
60,23,254,238,236,238,230,112,60,60,0
 1130VDU19,2,1;0;19,1,2;0;:Q%=21:T%=5
76:U%=572:A%(0)=13:B%(0)=5:E%(0)=1:A%
(1)=1:B%(1)=24:D%(1)=1:N$="SILLY STEP
S":PRINTTAB(19,22)K$TAB(11,20)K$TAB(1
,13)K$TAB(19,6)K$TAB(6,9)T$TAB(11,15)
T$
 1140RETURN
 1150VDU23,239,3,15,63,113,111,249,24
9,255,23,240,192,240,252,142,246,159,
159,255,23,241,239,223,219,104,124,62
,15,3,23,242,247,251,219,22,62,124,24
0,192
 1160VDU23,243,3,15,63,127,113,239,24
9,255,23,244,192,240,252,254,142,247,
159,255,23,245,255,247,239,124,127,63
,15,3,23,246,255,239,247,62,254,252,2
40,192
 1170VDU23,247,0,0,0,4,12,30,15,7,23,
248,0,0,0,0,0,0,6,142,23,249,3,0,3,15
,31,63,31,7,23,250,58,250,246,252,248
,240,192,0
 1180VDU23,251,0,0,0,0,0,0,96,113,23,
252,0,0,0,32,48,120,240,224,23,253,92
,95,111,63,31,15,3,0,23,254,192,0,192
,240,248,252,248,224
 1190VDU19,2,3;0;19,1,5;0;:A%(0)=9:B%
(0)=19:E%(0)=1:A%(1)=6:B%(1)=8:D%(1)=
1:N$="SUNSHINE CAVERNS":T%=1152:U%=41
2:Q%=22:PRINTTAB(0,18)K$TAB(12,12)K$T
AB(0,5)K$TAB(11,5)K$TAB(3,7)T$TAB(16,
15)T$
 1200RETURN
 1210VDU23,239,15,31,24,24,57,57,48,4
8,23,240,254,255,31,15,206,142,62,14,
23,241,112,115,103,103,224,224,255,12
7,23,242,12,204,156,156,24,24,248,240
 1220VDU23,243,15,31,24,24,57,57,48,4
8,23,244,254,255,31,15,206,142,62,14,
23,245,112,115,103,103,224,224,255,12
7,23,246,12,204,156,156,24,24,248,240
 1230VDU23,247,15,31,28,28,56,57,51,5
1,23,248,254,255,31,15,198,230,254,25
4,23,249,115,103,103,103,240,240,255,
127,23,250,252,252,204,156,24,56,248,
240
 1240VDU23,251,15,31,28,28,56,57,51,5
1,23,252,254,255,31,15,198,230,254,25
4,23,253,115,103,103,103,240,240,255,
127,23,254,252,252,204,156,24,56,248,
240
 1250VDU19,2,5;0;19,1,6;0;:A%(0)=7:B%
(0)=15:D%(0)=-1:A%(1)=11:B%(1)=15:D%(
1)=1:N$="ADVERTISING HALL":T%=1152:U%
=860:Q%=22:PRINTTAB(3,13)K$TAB(10,10)
K$TAB(17,13)K$TAB(1,5)K$TAB(3,29)T$"
 "T$"   "T$" "T$TAB(2,20)S$
 1260PRINTTAB(0,5)S$TAB(5,7)T$TAB(11,
7)T$TAB(13,5)S$:PRINTTAB(9,15)M$(0):R
```

→

YOUR MOVE...

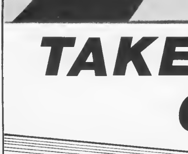
TAKE A PIECE OF OUR SOFTWARE!

SUPER GAMES

ONLY
£4.95
INC VAT
&
p+p

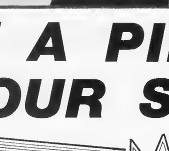
SUPER VALUE

ER*BERT



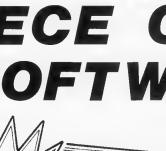
Electron & BBC 32K
Join Erbert in his cubic domain – fast and funny. Avoid his unwelcome guests. Many features – alternative screen displays – addictive!
MACHINE CODE GAME

3D SPACE RANGER



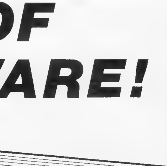
BBC 32K (OS 1.0 or 1.2)
Excellent 3D graphics four different scenes. Battle to the death star and destroy it.
MACHINE CODE GAME

PINBALL



Electron & BBC 32K
The classic arcade game up to four players with bonus features.
MACHINE CODE GAME

REVERSI



Electron & BBC 32K
A game of strategy and skill – 2 levels

MICROBYTE
SOFTWARE

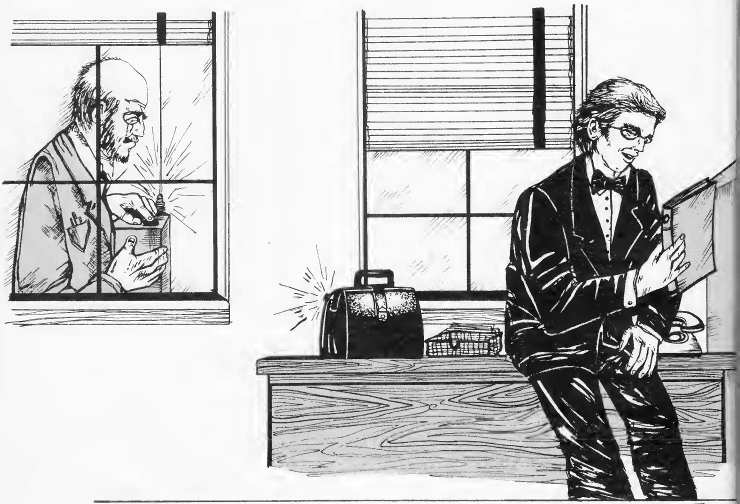
Games available now at many computer shops – or by fast mail order from Microbyte Software. S.A.E. for illustrated brochure. Trade enquiries welcome. Access 24 hour hot line **06373 6886.**

MICROBYTE SOFTWARE (Dept. E10)
18 Hilgrove Road, Newquay, Cornwall TR7 2QZ

From Page 7

```
ETURN
 1270VDU23,239,112,136,168,120,60,60,
30,30,23,240,14,17,21,30,60,60,120,12
0,23,241,63,58,114,120,61,31,50,50,23
,242,252,92,78,30,188,248,76,76
 1280VDU23,243,28,42,34,60,120,240,12
4,30,23,244,56,84,68,60,30,31,62,120,
23,245,63,63,114,120,63,31,13,13,23,2
46,252,252,78,30,252,248,176,176
 1290VDU23,247,3,7,7,76,126,63,15,7,2
3,248,192,224,241,159,190,248,240,240
,23,249,3,1,1,1,3,7,14,24,23,250,240,
224,224,192,192,128,0,0
 1300VDU23,251,3,7,143,249,125,31,15,
15,23,252,192,224,224,50,126,252,240,
224,23,253,15,7,7,3,3,1,0,0,23,254,19
2,128,128,128,192,224,112,24
 1310VDU19,2,6;0;19,1,1;0;:A%(0)=3:B%
(0)=20:D%(0)=1:A%(1)=5:B%(1)=16:D%(1)
=1:N$="SPOOKY SIDE-CHAMBER":T%=1088:U
%=636:Q%=18:PRINTTAB(19,27)K$TAB(7,19
)K$TAB(10,19)K$TAB(9,14)K$TAB(6,29)T$
"  "T$"  "T$"  "T$
 1320PRINTTAB(4,26)S$"  "S$"  "S$"  "
S$"  "S$TAB(6,11)T$TAB(12,11)T$:RETUR
N
 1330VDU23,239,1,3,7,15,31,25,53,49,2
3,240,128,192,224,240,248,248,252,236
,23,241,123,127,127,3,7,63,30,7,23,24
2,246,246,246,238,220,184,112,224
 1340VDU23,243,1,3,7,15,31,31,63,55,2
3,244,128,192,224,240,248,152,172,140
,23,245,111,111,111,119,59,29,14,7,23
,246,222,254,254,192,224,248,112,224
 1350VDU23,247,1,3,7,7,7,7,3,1,23,248
,128,192,224,160,160,96,192,128,23,24
9,1,15,17,1,7,4,12,0,23,250,240,128,1
28,128,224,32,32,48
 1360VDU23,251,1,3,3,3,3,3,1,17,23,25
2,128,192,192,64,64,192,128,128,23,25
3,15,1,1,1,7,4,4,12,23,254,128,240,13
6,128,224,32,48,0
 1370VDU19,2,3;0;19,1,4;0;:A%(0)=3:B%
(0)=9:E%(0)=1:A%(1)=14:B%(1)=9:D%(1)=
-1:N$="MINERS DINER":T%=0:U%=828:Q%=2
6:PRINTTAB(2,5)S$" "S$TAB(9,21)K$TAB(
16,11)K$TAB(10,7)K$TAB(3,5)K$TAB(12,2
9)T$
 1380RETURN
 1390VDU23,239,3,3,49,31,7,3,3,7,23,2
40,192,192,142,248,96,192,64,224,23,2
41,14,220,120,48,0,0,0,0,23,242,112,5
9,30,12,0,0,0,0
 1400VDU23,243,0,0,3,3,1,15,31,51,23,
244,0,0,192,192,128,240,120,204,23,24
5,99,71,30,56,48,24,56,112,23,246,70,
226,120,28,12,24,28,14
 1410VDU23,247,7,15,31,63,115,99,99,1
27,23,248,192,224,240,248,156,140,140
,252,23,249,61,63,28,16,21,31,15,3,23
,250,120,248,112,16,80,240,224,128
 1420VDU23,251,7,15,31,63,115,99,99,1
27,23,252,192,224,240,248,156,140,140
,252,23,253,61,63,31,24,15,3,0,0,23,2
54,120,248,240,48,224,128,0,0
 1430VDU19,2,1;0;19,1,3;0;:A%(0)=3:B%
(0)=19:D%(0)=1:A%(1)=11:B%(1)=17:E%(1
)=1:N$="ANCIENT ARCH":T%=1152:U%=764:
Q%=22:PRINTTAB(0,5)K$TAB(8,22)K$TAB(7
,17)K$TAB(7,8)K$TAB(4,22)S$TAB(5,26)T
$"   "T$"   "T$
 1440RETURN
 1450VDU23,239,0,0,0,0,0,7,15,14,23,2
40,48,120,120,60,6,195,230,44,23,241,
14,14,15,8,12,7,3,15,23,242,184,48,22
4,32,96,192,128,224
 1460VDU23,243,0,0,0,7,15,8,10,8,23,2
44,0,48,120,126,179,198,236,248,23,24
5,15,12,13,7,3,3,3,15,23,246,240,96,9
6,192,128,128,128,224
 1470VDU23,247,1,1,15,1,81,115,37,63,
23,248,128,128,240,128,138,206,164,25
2,23,249,2,3,0,54,0,94,64,27,23,250,6
4,192,0,216,2,122,0,108
 1480VDU23,251,1,1,3,97,97,35,37,63,2
3,252,128,128,192,134,134,196,164,252
,23,253,3,3,0,27,64,94,0,54,23,254,19
2,192,0,108,0,122,2,216
 1490VDU19,2,6;0;19,1,2;0;:A%(0)=3:B%
(0)=15:D%(0)=1:A%(1)=14:B%(1)=18:D%(1
)=-1:N$="RAMPAGING ROBOTS":T%=1024:U%
=796:Q%=22:PRINTTAB(2,5)K$TAB(5,7)K$T
AB(16,11)K$TAB(10,21)K$TAB(12,29)T$"
 "T$
 1500RETURN
 1510VDU23,239,16,16,24,8,12,12,6,6,2
3,240,16,16,48,32,96,96,192,192,23,24
1,3,1,3,6,14,14,6,2,23,242,128,0,128,
192,224,224,192,128
 1520VDU23,243,0,0,64,96,48,56,28,14,
23,244,0,0,4,12,24,56,112,224,23,245,
3,1,3,14,12,24,24,8,23,246,128,0,128,
224,96,48,48,32
 1530VDU23,247,0,0,0,252,252,104,120,
108,23,248,0,0,0,0,0,0,60,230,23,249,
127,117,255,255,255,0,248,248,23,250,
242,251,255,254,248,3,127,30
 1540VDU23,251,252,252,104,120,108,12
7,117,255,23,252,0,0,0,60,230,242,251
,255,23,253,255,255,0,248,248,0,0,0,2
3,254,254,248,3,127,30,0,0,0
 1550VDU19,2,3;0;19,1,1;0;:A%(0)=3:B%
(0)=5:E%(0)=1:A%(1)=3:B%(1)=17:D%(1)=
1:N$="ESCAPED EQUIPMENT":T%=1152:U%=1
24:Q%=26:PRINTTAB(6,20)S$" "S$TAB(14,
21)S$" "S$TAB(7,20)K$TAB(15,21)K$TAB(
0,5)K$TAB(8,7)K$TAB(18,24)T$
 1560PRINTTAB(6,15)T$:RETURN
 1570VDU23,239,3,15,63,255,1,1,1,1,23
,240,192,240,252,255,128,128,128,128,
23,241,127,15,39,112,219,219,115,32,2
3,242,254,240,228,14,219,219,206,4
 1580VDU23,243,0,0,0,3,15,63,255,1,23
,244,0,0,0,192,240,252,255,128,23,245
,127,15,39,112,219,219,115,32,23,246,
254,240,228,14,219,219,206,4
 1590VDU23,247,0,0,0,0,0,0,0,25,23,24
8,24,60,126,231,195,231,126,188,23,24
9,61,126,231,195,231,126,60,24,23,250
,152,0,0,0,0,0,0,0
 1600VDU23,251,24,60,126,231,195,231,
126,61,23,252,0,0,0,0,0,0,0,152,23,25
3,25,0,0,0,0,0,0,0,23,254,188,126,231
,195,231,126,60,24
 1610VDU19,2,2;0;19,1,4;0;:A%(0)=17:B
%(0)=19:D%(0)=-1:A%(1)=10:B%(1)=11:D%
(1)=1:N$="PROCESSING PLANT":T%=0:U%=7
64:Q%=19:PRINTTAB(4,18)STRING$(7,S$)T
AB(14,18)S$S$S$TAB(5,16)STRING$(4,T$+
"  ")TAB(5,7)T$
 1620PRINTTAB(3,12)T$"  "T$TAB(10,29)
T$TAB(3,18)K$TAB(6,18)K$TAB(17,18)K$T
AB(5,5)K$:RETURN
 1630VDU23,239,1,3,3,6,5,12,63,255,23
,240,128,192,192,96,160,48,252,255,23
,241,234,106,63,31,15,9,16,0,23,242,8
7,86,252,248,224,144,8,0
 1640VDU23,243,1,3,3,7,7,12,63,255,23
,244,128,192,192,224,224,48,252,255,2
3,245,245,117,63,31,7,25,0,0,23,246,1
75,174,252,248,224,152,0,0
 1650VDU23,247,0,0,0,4,34,2,0,24,23,2
48,0,0,0,36,64,128,152,36,23,249,4,2,
0,0,3,3,3,3,23,250,64,0,8,0,192,192,1
92,192
 1660VDU23,251,0,0,0,0,12,16,32,1,23,
252,0,0,0,4,0,16,8,8,23,253,0,0,32,64
,67,67,3,3,23,254,8,0,32,0,196,196,19
6,192
 1670VDU19,2,3;0;19,1,5;0;:A%(0)=5:B%
(0)=9:D%(0)=1:A%(1)=13:B%(1)=9:D%(1)=
-1:N$="THE GRAND FINALE":T%=0:U%=764:
Q%=23:PRINTTAB(12,6)K$TAB(11,12)K$TAB
(3,19)K$TAB(17,19)K$TAB(4,18)T$TAB(13
,22)T$TAB(15,29)T$
 1680PRINTTAB(14,19)S$TAB(10,12)S$TAB
(2,29)T$" "T$TAB(3,14)CHR$235:RETURN
 1690MODE7:FORI%=1TO4:SOUND1,1,40,12:
NEXT:PROCdbl("CONGRATULATIONS",2):PRO
Cdbl("YOU HAVE ESCAPED",5):PROCdbl("T
HE LOST URANIUM MINE",8):PROCdbl("WIT
H A FORTUNE",11):PROCdbl("IN GOLD KEY
S !",14)
 1700SCR%=SCR%+5000*LIV%:PROCdbl("BON
US "+STR$(5000*LIV%),18):FI%=1:FORI=1
TO17500:NEXT:RETURN
 1710REM
 1720REM #### EXPLORER EDDIE ####
 1730REM
 1740REM  #### BY B.WAKELIN ####
```

If you're devious, deft and deadly – go



PROCEDURES

PROCtitle	Sets up MasterSpy title.
PROCerr	Error trap. If ERR=17 PROCreport ELSE REPORT ERROR with my name and address.
PROCinit	Sets DIM arrays for Messages (Mes$), Colours (Col%), Answers (Ans%), Codes (Code$) and Random (rnd%).
PROCagent	Asks for the player's name, waits for a mission.
PROCinstruction	Gives briefing to agent.
PROCscore	Keeps score.
PROCsetcode	Sets all codes, and makes sure that Files 1-3 do not duplicate any colour.
PROCplayingboard	Sets screen up for play.
PROCplay	Prints colour key and changes numbered input to the right coloured ★. Keeps tries to 10 per file.
PROCreport	Gives a mission report.
PROCcheck	Checks answer and assigns clues.
PROCans	Reveals correct code.
PROCnewpage	Press space bar to continue.
PROCclues	Mixes up answers and prints them as on screen '/ X .'
PROCnewsflash	If score is too good a newsflash from mission control.
PROCscoretable	There are five places on the agent's league table. You have to be good to beat James Bond.
PROCinsert	If an error other than 17(Escape) then this will list the line in which the error has occurred.

VARIABLES

HI$(5)	James Bond.
H% (5)	Minimum score to enter score table.
level%(6)	File level in play.
attempts%	Total goes, all files.
of%	Number of files opened.
err%	Flag. . err%=1 or 2. If 1 program terminated. If 2 program completed so that correct message is printed.
Cn%	Number of colours in code (1-5).
N%	Number of tries in each file (max 10).
M%	Counts how many '/' (right colour right place) in your answer. If M%=Cn% it is correct.
Tries%	Puts number of tries on screen.
Col%	Colours used in each file.

on a mission with JOHN HOLLINGS . . .

But you have to be good to beat James Bond!

MASTERSPY is based on the well-known colour code game Mastermind. In this version you are a secret agent who, once accepted into the agency, is sent on a mission by mission control. You must uncover the secret colour codes of six files – but it is not easy.

File 1 contains any three of seven colours. File 2 contains any four of seven colours. File 3 contains any five of seven colours. These files do not contain any duplication of colours.

Once you have opened File 3 (five colours) you get on to the more difficult agency work. The next three files, although reverting to three, four and five colours of seven respectively, now may contain any number of colour duplications.

As you have to be good to be an agent you are only given 10 attempts at each file. If you don't crack by the tenth try the code will be revealed to you and you lose the file you previously opened, so you go back by one file. On File 1 should you fail to uncover the code it is rescrambled and you can try again.

During your spying activities mission control keeps an undercover watch over your progress. If M decides you are likely to be caught he instructs that your mission is terminated and you will receive a mission report to that effect.

If on the other hand you complete your mission by opening all six files you will receive a report from your mission controller and may be duly promoted.

To join the ranks of James Bond you have to be very good. You must open all six files in fewer than 50 attempts. If you can do this you are invited to enter your name on the agents' league table.

To help you in your mission your controller gives you clues for each colour entered – but these do not relate to the correct position of the colours in the code. To enter your selection you must depress the relevant number key followed by Return for each colour in the code.

When the program runs you enter your name and if accepted for a mission you are asked: "Do you require a briefing?". This is a must for a newly enrolled agent as it contains mission control's clues, which must be memorised.

Good luck in your mission.

```
    5 REM (C) The Micro User
   10 MODE7
   20 *FX200,1
   30 VDU23;8202;0;0;0;:CLS:PROCtitle
   40 CLEAR
   50 *FX4,1
   60 *KEY10 O.|MSOUND1,-15,75,5:SOUN
D2,-15,100,5:|M RUN |M
   70 REM ON ERROR PROCerr
   80 VDU23;8202;0;0;0;
   90 DIM HI$(5),H%(5),P%(5):FORA=1TO
5:HI$(A)="James Bond"+STRING$(13,".")
:H%(A)=50:P%(A)=A:NEXT A
  100 PROCinit
  110 level%=1:attempts%=0:of%=0:err%
=0:file1%=0:file2%=0:file3%=0:file4%=
0:file5%=0:file6%=0
  120 PROCtitle:PROCagent:PROCtitle:P
RINTTAB(13,8);CHR$131;"Agent ";Name$
  130 PRINTTAB(2,10)CHR$131;" you hav
e been accepted for a mission"
  140 REPEATPRINTTAB(4,12)CHR$131;"Do
 you require a briefing? (Y/N)";:Z$=G
ET$
  150 X%=INSTR("YyNn",Z$)
  160 UNTIL (X%<>0)
  170 IF X%=1 OR X%=2 THEN PROCinstru
ctions ELSE 180
  180 IF level%>6 THEN err%=2:of%=6:C
LS:PROCscore
  190 IF level%=6 OR level%>3 THEN Cn
%=level%-1 ELSE Cn%=level%+2
  200 M%=0:N%=0:X=2:Y=6:Tries%=0:VDU2
3;8202;0;0;0;
  210 PROCsetcode
  220 PROCtitle:PROCplayingboard
  230 REPEAT
  240 Tries%=Tries%+1
  250 N%=N%+1:Y=Y+1
  260 IF M%=Cn% THEN 360
  270 PROCplay
  280 attempts%=attempts%+1
  290 IF of%<5 AND attempts%>50 THEN
300 ELSE 320
  300 err%=1:CLS:PROCreport:SOUND1,1,
136,50:ENVELOPE1,1,-7,7,0,10,10,0,126
,0,0,-126,126,126:Z=INKEY(200)
  310 PROCtitle:PROCreport:PROCplayin
gboard:PROCans:PROCscore
  320 UNTIL N%=10:IF M%=Cn% THEN 360
  330 PROCans:PRINTTAB(0,23);SPC(39);
:PRINTTAB(11,23);CHR$133;"Incorrect c
ode ":VDU7:Z=INKEY(200)
  340 PROCnewpage
  350 IF level%=1 THEN of%=0:GOTO 180
 ELSE level%=level%-1:of%=level%:GOTO
 180
  360 PROCans:PRINTTAB(0,23);SPC(39);
:PRINTTAB(12,23);CHR$131;"Correct cod
e ":SOUND1,2,4,50:ENVELOPE2,2,6,0,0,2
55,0,0,126,0,0,-126,126,126:Z=INKEY(2
50)
  370 of%=level%-1:level%=level%+1:GO
TO 180
  380 END
  390 REM ****** init ******
  400 DEFPROCinit
  410 DIM Mes$(23):FORA=1TO23:Mes$(A)
=" ":NEXT
  420 DIM Col%(5),Ans%(5),Code$(5),rn
d%(5),A(7)
  430 A=0:REPEAT:A=A+1
  440 Col%(A)=0:Ans%(A)=0:Code$(A)=".
":UNTIL A=5
  450 filec%=135:filel%=129
  460 ENDPROC
  470 REM ****** setcode ******
  480 DEFPROCsetcode
  490 IF level%<=3 THEN 530 ELSE 500
  500 C%=0:REPEAT:C%=C%+1
  510 Col%(C%)=RND(7)
  520 UNTIL C%=Cn%:ENDPROC
  530 FOR Z=1TO7:A(Z)=0:NEXT
  540 C=0:Q=0:REPEAT:Q=Q+1:IF C=7 END
PROC
  550 X=RND(-TIME)
  560 X=RND(7)
  570 IF A(X)=1 THEN Q=Q-1:GOTO 590 E
LSE A(X)=1
  580 Col%(Q)=X
  590 UNTIL Q=Cn%
  600 ENDPROC
```

→

From Page 11

```
 610 REM ***** playingboard  *****
 620 DEFPROCplayingboard
 630 PR1NTTA8(11,3);CHR$(filec%);CHR
$157;CHR$(file1%);CHR$136;"F I L E  "
;level%;"  "CHR$156
 640 PR1NTTA8(0,5);CHR$132;CHR$157;C
HR$131;"Selection";TAB(17)"Tries";TAB
(31)"Clues         "CHR$156
 650 ENDPROC
 660 REM ****** play ******
 670 DEFPROCplay
 680 *FX15,0
 690 PRINTTAB(13,19);CHR$134;"Colour
 key"
 700 A%=0:A=128:x%=10:REPEAT:A%=A%+1
:A=A+1:x%=x%+2
 710 PR1NTTAB(x%,20);CHR$(A);CHR$255
;" "
 720 UNT1L A%=7
 730 N=0:x1%=11:REPEAT:N=N+1:x1%=x1%
+2
 740 PR1NTTAB(x1%,21);N
 750 UNTIL N=7
 760 PR1NTTAB(0,23);SPC(39);
 770 A=0:X=18:REPEAT:A=A+1:X=X+2
 780 PR1NTTAB(0,23);CHR$131;"Enter "
;Cn%;" colour code"
 790 VDU21
 800 INPUTTAB(18,23),Ans%(A)
 810 VDU6
 820 IF Ans%(A)<1 OR Ans%(A)>7 THEN
SOUND1,-15,75,5:GOTO 780
 830 VDU6:PR1NTTAB(X,23)CHR$(Ans%(A)
+128);"*";" "
 840 UNTIL A=Cn%:Q=1NKEY(50)
 850 A=5-Cn%:B=0:REPEAT:A=A+2:B=B+1:
PR1NTTAB(A,Y);CHR$(Ans%(B)+128);"*";:
UNTIL B=Cn%
 860 PROCcheck
 870 PR1NTTAB(18,Y);CHR$135;Tries%;
 880 PROCclues
 890 ENDPROC
 900 REM ****** check ******
 910 DEFPROCcheck:M%=0
 920 FOR A=1 TO Cn%
 930 1F Ans%(A)=Col%(A) THEN Code$(A
)="/":M%=M%+1:NEXT A:GOTO 980 ELSE
 940 B=0:REPEAT:B=B+1
 950 1F Ans%(A)=Col%(8) THEN Code$(A
)="X":GOTO 960 ELSE
 960 UNT1LB=Cn%
 970 NEXT A
 980 ENDPROC
 990 REM ****** ans ******
1000 DEFPROCans
1010 PR1NTTAB(5,3);SPC(20);:A=0:T=17
-Cn%
1020 REPEAT:A=A+1:T=T+2
1030 PR1NTTAB(T,3);CHR$(Col%(A)+128)
+"*";" ":UNT1L A=Cn%
1040 ENDPROC
1050 REM ****** clues ******
1060 DEFPROCclues
1070 xx%=31-Cn%:REPEAT:xx%=xx%+2
1080 X=RND(Cn%)
1090  IF rnd%(X)=1 THEN xx%=xx%-2:GO
TO 1120 ELSE rnd%(X)=1
1100 PRINTTAB(xx%,Y);CHR$131;Code$(X
)
1110 C=C+1
1120 UNTIL C=Cn%
1130 C=0
1140 FORA=1TOCn%:rnd%(A)=0:NEXT
1150 X=RND(Cn%)
1160 FORA=1TOCn%:rnd%(X)=0:NEXT
1170 FOR A=1 TO Cn%:Ans%(A)=0:Code$(
A)=".":NEXT
1180 ENDPROC
1190 REM ******* score ******
1200 VDU6
1210 DEFPROCscore
1220 *FX15,1
1230 IF attempts%<=30 AND of%=6 PROC
newsflash:BOTO 1570
1240 IF err%=1 THEN 1260
1250 FORh%=10TO11:PRINTTAB(10,h%);CH
R$130;CHR$136;CHR$141;"Mission Comple
ted":NEXT
1260 Z=INKEY(300):CLS
1270 FORh%=0TO1:PR1NTTAB(10,h%);CHR$
131;CHR$141;"Mission Report":NEXT
1280 RESTORE 2570:A=0:REPEAT:A=A+1
1290 READ Mes$(A)
1300 UNTIL A=23
1310 PR1NTTAB(0,4);CHR$134;"Ref Agen
t:";CHR$134;Name$
1320  PRINTTA8(10,6);CHR$134;"Classi
fied";CHR$133;"Secret"
1330 PR1NTTAB(0,9);;CHR$134;Mes$(1);
:IF err%=1 THENPR1NTTA8(20,9);;CHR$13
3;Mes$(2);;CHR$134;Mes$(4) ELSE PR1NT
TAB(20,9);;CHR$130;Mes$(3);;CHR$134;M
es$(4)
1340 PR1NTTAB(0,10);;CHR$134;Mes$(5)
;CHR$131;of%;;CHR$134;Mes$(6)
1350 PR1NTTAB(0,11);;CHR$134;Mes$(7)
:PR1NTTAB(4,11);CHR$131;attempts%;;CH
R$134;Mes$(8)
1360 PR1NTTAB(0,12);;CHR$134;;CHR$13
4;Mes$(9)
1370 IF err%=1 THEN PR1NTTAB(17,12);
;CHR$133;Mes$(10);;CHR$134;Mes$(12) E
LSE PR1NTTA8(17,12);;CHR$130;Mes$(11)
;;CHR$134;;CHR$134;Mes$(12)
1380 PR1NTTA8(0,13);CHR$134;Mes$(13)
1390 1F of%<6 THEN 1400 ELSE GOTO 14
40
1400 d=RND(3)
1410 IF d=1 THEN D$=Mes$(14):GOTO 14
80
1420 IF d=2 THEN D$=Mes$(15):GOTO 14
80
1430 IF d=3 THEN D$=Mes$(16):GOTO 14
80
1440 IF of%=6 AND attempts%>80 THEN
D$=Mes$(17):GOTO 1480
1450 1F of%=6 AND attempts%<71 AND a
ttempts%>60 THEN D$=Mes$(18):GOTO 148
0
1460 IF of%=6 AND attempts%<61 AND a
ttempts%>50 THEN D$=Mes$(19):GOTO 148
0
1470 1F attempts%<50 THEN D$=Mes$(20
):GOTO 1480
1480 PRINTTAB(3,13);;CHR$131;D$
1490 PR1NTTAB(0,14);;CHR$134;Mes$(21
)
1500 PR1NTTA8(6,17);CHR$131;Mes$(22)
1510 PRINTTAB(0,19);;CHR$129;;CHR$13
6;Mes$(23)
1520 Z=INKEY(2000)
1530 CLS:PROCtitle
1540 FORh%=10TO11:PRINTTAB(8,h%);CHR
$131;CHR$136;CHR$141;"Message destroy
ed":NEXT
1550 Z=INKEY(200)
1560 PROCscoretable
1570 REPEAT:PR1NTTAB(4,21);CHR$130;"
There's another mission for you"'TA8(
10);CHR$130;"Do you accept Y/N";:Z$=G
ET$
1580 Z%=1NSTR("YyNn",Z$)
1590 UNT1L (Z%<>0)
1600 IF Z%=1 OR Z%=2 THEN GOTO 110 E
LSE CLS:PR1NT"8asic":END
1610 END
1620 REM ****** agent *****
1630 DEFPROCagent
1640 PROCtitle
1650 PRINTTAB(0,4);CHR$134;"8efore y
ou can become an Agent you must";CHR$
134;"state your name for our files. T
hen you ";CHR$134;"will be considered
 for a mission."
1660
1670 PRINTTA8(1,10);CHR$131;"Please
state your name."
1680 1NPUTTAB(27,10)Name$:Name$=LEFT
$(Name$,18)
1690 PR1NTTAB(1,13);CHR$130;Name$;"
you are being considered"
1700 Z=1NKEY(400)
1710 ENDPROC
1720 REM ****** title ******
1730 DEFPROCtitle:CLS
1740 FOR h%=0 TO1:PR1NTTAB(7,h%)CHR$
129;CHR$157;CHR$135;CHR$141" M a s t
e r S p y   "CHR$156:NEXT h%
1750 ENDPROC
1760 REM ****** report ******
1770 DEFPROCreport
```

```
 1780 FORh%=IOTO11:PRINTTAB(10,h%);CH
R$131;CHR$136;CHR$141;"Mission Report
":NEXT
 1790 ENDPROC
 1800 REM ****** newsflash ******
 I810 DEFPROCnewsflash
 1820 Name$="john":attempts%=23
 1830 CLS:FOR h%=10 TO11:PRINTTAB(I2,
h%)CHR$141;CHR$131;CHR$136;"NEWSFLASH
":NEXT h%
 1840 Z=INKEY(200):CLS
 1850 PROCtitle
 1860 PRINTTAB(13,3);CHR$131;CHR$136;
"Newsflash"
 1870 PRINTTAB(0,5);CHR$134;"Ref: Age
nt";CHR$131;Name$
 1880  PRINTTAB(0,7);CHR$134;"Congrat
ulation You opened all 6 files"';CHR$
134;"in only ";attempts%" attempts.
How ever 'M' is"';CHR$I34;"puzzled as
 it is almost impossible to"';CHR$134
;"do that. Therefore you are";CHR$133
;"Dismissed"';
 1890 PRINTCHR$134;"from the service
immediately."
 1900 Z=INKEY(200):PRINT'';CHR$131;"H
ow ever 'M' thinks you should have"'T
AB(12)CHR$131;"a second chance":GOTO1
570
 1910 END
 1920 REM ****** score table ******
 1930 DEFPROCscoretable
 1940 PROCtitle:PRINTTAB(9,3);CHR$134
;"Agent's League Table":PRINTTAB(9,4)
;CHR$134;STRING$(20,"-")
 1950 FOR N%=1 TO 5
 1960 PRINTTAB(5,(N%*2)+6);CHR$134;P%
(N%);CHR$131;H1$(N%);H%(N%):NEXT
 1970 COLOUR3:IF attempts%>H%(5) OR a
ttempts%<6 ENDPROC
 1980 N%=-1:REPEAT:N%=N%+1:UNTIL atte
mpts%<H%(N%) OR N%=5:IF N%=5 OR err%=
1 ENDPROC
 1990 H%(N%)=attempts%:PRINTTAB(12,19
);:VDU141,136:PRINTCHR$134;"Hi-score
!!!";TAB(12,20);:VDU141,136:PRINTCHR$
134;"Hi-score !!!"
 2000 PRINT'TAB(12);CHR$134;"Enter yo
ur name-"''TAB(13);
 2010 VDU31,32:INPUTA$:IF LEN(A$)>18
A$=LEFT$(A$,18):REPEAT:A$=A$+".":UNTI
L LEN(A$)=24 ELSE REPEAT:A$=A$+".":UN
TIL LEN(A$)=23
 2020 H1$(N%)=A$:attempts%=100:CLS:GO
TO 1940
 2030 ENDPROC
 2040 REM ****** instructions ******
 2050 DEFPROCinstructions
 2060  PROCtitle
 2070 PRINTTAB(0,4);CHR$130;"Briefing
 for Agent ";Name$;
 2080  PRINT''"  Your mission is to g
ain access to 6"'"  files held in the
 Udanddy Embassy."
 2090  PRINT'"  Each file is in a sep
arate safe with"'"  its own colour-co
de lock and is graded  by the classif
ication of the file it"'"  is protect
ing."
 2100 PRINT';CHR$131;"File No."TAB(12
)"Colours in the code-lock"
 2110 PRINT'TAB(2);CHR$I34;"1"TAB(10)
"3 colours No duplication"
 2I20 PRINTTAB(2);CHR$134;"2"TAB(10)"
4 colours No duplication"
 2130 PRINTTAB(2);CHR$134;"3"TAB(10)"
5 colours No duplication"
 2140 PRINTTAB(2);CHR$134;"4"TAB(10)"
3 colours May be duplicated"
 2150 PRINTTAB(2);CHR$134;"5"TAB(10)"
4 colours May be duplicated"
 2160 PRINTTAB(2);CHR$134;"6"TAB(10)"
5 colours May be duplicated"
 2170 PROCnewpage
 2180 CLS:PROCtitle
 2190 PRINTTAB(0,4);CHR$130;"Briefing
 for Agent ";Name$;
 2200 PRINT''" The colours to choose
from are as shown below together with
 their appropriate"'" number-key."
 2210 A%=0:A=I28:x%=10:REPEAT:A%=A%+1
:A=A+1:x%=x%+2
 2220 PRINTTAB(x%,11);CHR$(A);CHR$255
;" ";
 2230 UNTILA%=7
 2240 N=0:x%=11:REPEAT:N=N+1:x%=x%+2
 2250 PRINTTAB(x%,12);N:UNTIL N=7
 2260 PRINT''" You are given 10 tries
 at each code. If you do not open a f
ile by the 10th try  the code is reve
aled, but you go BACK a file.  If you
 are taking too many goes  for the nu
mber of open files or if it"'" looks
like you could";
 2270 PRINT" be caught then the missi
on controller will terminate your  mi
ssion."
 2280 PROCnewpage
 2290 PROCtitle
 2300 PRINTTAB(0,4);CHR$130;"Briefing
 for Agent ";Name$;
 2310 PRINT''" There are clues to hel
p you crack the   codes. Memorize the
 following."'';CHR$131;"     / = righ
t colour right place"';CHR$131;"
X = right colour wrong place"';CHR$13
1;"     . = colour not used."
 2320 PRINT'" However these clues are
 mixed up e.g.   clue 1 may not relat
e to colour 1, etc. If you are good y
ou can be promoted."
 2330 PRINT'" That concludes the brie
fing for this"'" mission ";Name$;'TAB
(15);"Good luck"
 2340 PRINT'TAB(7);CHR$131;"Signed Mi
ssion controller"
 2350 PROCnewpage
 2360 ENDPROC
 2370 END
 2380 REM ****** newpage ******
 2390 DEFPROCnewpage
 2400 PRINTTAB(3,23);CHR$130;"Press t
he <Space-bar> to continue"
 2410 Z$=GET$:IFZ$<>" " VDU7:GOTO 240
0
 2420 ENDPROC
 2430 REM ****** err ******
 2440 DEFPROCerr:VDU6:CLS
 2450 SOUND1,-15,75,5:SOUND2,-15,100,
5
 2460 IF ERR=17 THEN err%=1:PROCrepor
t:PROCscore:END
 2470 *FX4,0
 2480 CLS:PRINT:REPORT:PRINT" at line
 No. ";ERL
 2490 PRINT';CHR$131;"Dear Player"'';
CHR$131;" I am sorry that an error ha
s appeared"';CHR$131;"in the program.
  If you cannot correct"';CHR$131;"it
 please let me know.   Tell me the"';
CHR$131;"error and the line number in
 which it"';
 2500 PRINTCHR$131;"occurred."';CHR$1
31;"Write to me at this address:-"
 2510 PRINT'TAB(12);CHR$134;"Mr J.HoI
lings"'TAB(12);CHR$134;"156 TempIewoo
d"'TAB(12);CHR$134;"Walter Ash"'TAB(1
2);CHR$134;"High Wycombe"'TAB(12);CHR
$134;"Bucks."
 2520 PRINT:PROCinsert("LIST"+STR$(ER
L)+CHR$13):STOP
 2530 END
 2540 DEFPROCinsert(I$):REPEAT:A%=138
:X%=0:Y%=ASC(LEFT$(I$,1)):CALL&FFFF4:
I$=RIGHT$(I$,LEN(I$)-1):UNTIL I$="":E
NDPROC
 2550 END
 2560 REM ****** DATA *******
 2570 DATA"As your mission was","Term
inated","Completed","and the","number
 of files opened was","which took","y
ou"," attempts. It is the decision of
"
 2580 DATA"'M' that you are","Demoted
","Promoted","to the rank","of","Clea
ner (at Hoover's)","Supervisor (in a
Siberian Salt mine)","Builder (for Le
go)","Assistant Spy","Spy","Super Spy
","Master Spy"
 2590 DATA"This is effective immediat
ely.","Signed Mission Controller","Th
is message self-destructs in 30 Sec"
 2600 END
```

Draw up a chair and join in this poker simulation game by ALAN GORNALL

Deal your way to a right royal flush

DRAW is a computer simulation of draw poker, from which all other forms of the game are derived.

First each player chips in his or her ante or stake into the kitty or pot. Then each is dealt five cards face down.

In real life the player to the dealer's left either opens the betting or folds, so losing his ante. The next player in turn may "see" the amount previously put into the pot, or raise it, or fold. And so on until we get back to the dealer.

If the pot has been raised the betting goes around a second time. Next comes the draw.

Each player can change any or all of the five cards in his hand. Once everyone has been dealt the final round of betting follows.

The same process as before applies and the last person left in collects the pot.

Poker hands are ranked like this, with the best hand first:

Royal flush (ace to ten in the same suit), straight flush (five cards in sequence of the same suit), four of a kind (say, four kings), full house (three cards of one kind and two of another), Flush (all of the same suite), Straight (consecutive cards, different suits), three of a kind (say three fives), two pairs, one pair.

The simulation, however, involves only two players with the computer acting as dealer. So if one player drops out the hand will be over.

A few tips for the player:

The only way to play poker, other than by cheating – which incidentally the simulation is quite adept at when called upon – is to make the most out of your winning hands, and lose as little as possible on your losing ones. Unfortunately, this is hard do to.

One way is to glean as much as possible from your opponent's style of play and act accordingly.

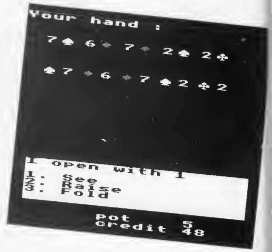
Of course he may play erratically, which would not be much help.

You will be pleased to hear that the simulation does play to a fixed system, the nature of which you'll have to work out for yourself.

There are two pretty good indicators as to the strength of an opponent's hand. One is how much money he is prepared to put on it – although he may bluff – and the other is how many cards he takes at the draw.

If he takes three cards, the best hand he could have had before the draw would be a pair, and his odds of improving this are about 5-2 against. So if you have two pairs, chances are you'll beat him in the showdown.

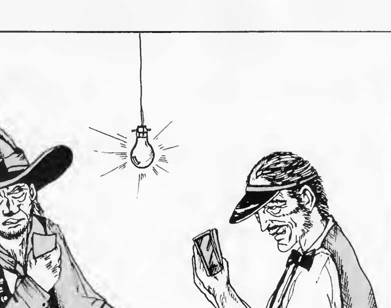
In the game the ante is 1, the maximum open 10 and the maximum raise 10.



```
Your hand :
  7♠ 6♦ 7♦ 2♠ 2♣

  ♠7 ♦6 ♦7 ♠2 ♣2



I open with 1
1: See
2: Raise
3: Fold

           pot      5
           credit  48
```

PROCEDURES

PROCvalue(Z$)	Values any poker hand in order of the accepted ranks. This uses other procedures, such as: PROCstraight, PROCswap, PROCflush, PROCpremanip, PROCa (deals with hands of one pair), PROCflush2, PROCb (hands of two pairs), PROCc (three of a kind), PROCd (full house), PROCflush1, PROCit, PROCe (four of a kind).
PROCdeal(HAND$,Y%)	Deals the specified hand. Starting at the left of the screen at a Y – coordinate of Y%. This uses PROCcard(X%,Y%) to draw the card in the correct place.
PROCval	Draws the value of the card on its face.
PROCinfo	Displays the value in the pot, and how much you are in credit.
PROCbet PROCbet1	Control the betting rounds. Which one is used depends on whose turn it is to open first.
PROCpre_bet PROCpre_bet1	Decide the computer betting scheme. The former is used before the draw has taken place, and the latter after it has taken place.
PROCbluff	Works out the percentage of hands that the computer's opponent has bluffed on.
PROCwin	Called when someone has been seen after the draw has taken place.
PROCdraw	Controls which cards you wish to change, and how many the computer wishes to change.
PROCcheat	Called when the name entered at the beginning of the program matches up with one of those in one of the data statements at the end of the program. It also sets its betting scheme, and if its hand is not as good as its opponent's hand it will alter it using PROCalter, and if it still isn't as good, it will change it using PROCalter1.



VARIABLES

Pack$ Contains a pack of cards.
STAKE% Opponent's credit.
POT% Amount of money currently in the pot.
bluff% Percentage of hands that the opponent has bluffed on.
no% Number of hands that have so far been played, with nine added on, to make calculations easier.
BLUFF% Number of hands that the computer's opponent has bluffed on.
Opponent$ Name of the computer's opponent.
FOLD% Variable by which computer decides whether it will fold or not, if 1, it will fold, if 0, it will not.
cheat% Variable by which computer decides whether it will cheat or not, if 1, it will cheat, if 0, it will not.
opt% What the computer's opponent has decided to do.
draw% Whether the draw has taken place or not, if 1, it has, if 0, it has not.
Count% Card at the top of the pack.
A$(52) After the cards have been shuffled they are placed here, one in each location.
Hand1$ Opponent's hand.
Hand2$ Computer's hand.
fold% Determines who has the write to fold and not lose their ante. If 1, it is the computer's turn, if 0, it is the opponent's turn.
value The value of the hand last used in PROCvalue.

All of the other variables are used in FOR . . . NEXT loops, are restricted to a small part of the program, or are of little importance.

```
   10 REM Draw-Poker by Alan Gornall
   20 REM Copyright Micro User
   30 REM 1984
   80 ON ERROR GOTO 5650
   90 MODE 5
  100 VDUI9,2,2,0,0,0
  110 VDU 23,240,8,28,28,107,127,107,
8,28
  I20 VDU 23,24I,8,28,62,I27,62,28,8,
0
  I30 VDU 23,242,54,127,127,I27,62,28
,8,0
  140 VDU 23,243,8,28,62,127,127,I27,
28,62
  150 DIM ST$(4)
  160 ST$(1)=CHR$ 18+CHR$ 0+CHR$ 0+CH
R$ 240
  I70 ST$(2)=CHR$ 18+CHR$ 0+CHR$ 1+CH
R$ 241
  180 ST$(3)=CHR$ I8+CHR$ 0+CHR$ I+CH
R$ 242
  190 ST$(4)=CHR$ 18+CHR$ 0+CHR$ 0+CH
R$ 243
  200 DIM YY$(3),A$(52),Z$(5),type%(1
4),local%(2),Z%(5),C%(52),A%(4,5)
  2I0 STAKE%=50:bet%=0:raise%=0:bluff
%=FALSE:no%=I0:BLUFF%=I:POT%=0
  220 Pack$="AC2C3C4C5C6C7C8C9CTCJCQC
KCAD2D3D4D5D6D7D8D9DTDJDQDKDAH2H3H4H5
H6H7H8H9HTHJHQHKHAS2S3S4S5S6S7S8S9STS
JSQSKS"
  230
  240 PRINT"What is your name":INPUT
Opponent$
  250 REPEAT
  260 READ F$
  270 UNTIL F$="****"OR F$=Opponent$
  280 IF F$=Opponent$ THEN cheat%=I E
LSE cheat%=FALSE
  290
  300 REPEAT
  310 IF STAKE% > 500 THEN cheat%=I
  320 fold%=FALSE
  330 REPEAT
  340 FOLD%=FALSE:flush%=FALSE:BLOP%=
FALSE:FOLDED%=FALSE:taken%=FALSE:draw
%=FALSE:stiff%=0:run%=FALSE:ace%=FALS
E:position%=FALSE
  350 CLS:PRINT TAB(2,26);"Shuffling
Cards"
  360 FOR A%= 1 TO 52
  370 C%(A%)=FALSE
  380 NEXT
  390 SEED%=RND(-TIME)
  400 FORA%=ITO52
  4I0 REPEAT
  420 B%=RND(52)
  430 UNTIL C%(B%) = FALSE
  440 E%=B%*2-I:C%(B%)=C%(B%)+1:A$(A%
)=MID$(Pack$,E%,2):NEXT
  450 HandI$=A$(1)+A$(3)+A$(5)+A$(7)+
```

⟶

From Page 15

```
A$(9)
  460 Hand2$=A$(2)+A$(4)+A$(6)+A$(8)+
A$(10)
  470 Count%=11
  480 STAKE%=STAKE%-1:POT%=POT%+2
  490 abo%=FALSE
  500
  510 COLOUR 130
  520 CLS
  530 PRINT:COLOUR0
  540 PRINT"Your hand :"
  550
  560 PROCdeal(Hand1$,700)
  570
  580 PROCvalue(Hand2$)
  590
  600 PROCinfo
  610 IF fold% THEN 3380 ELSE 3470
  620 PROCinfo
  630
  640 no%=no%+1
  650
  660 IF fold% THEN PROCbet ELSE PROC
bet1
  670
  680 IF (FOLD%=1 AND cheat%=FALSE) O
R (FOLDED% = 1 AND fold%=0) THEN 800
  690 IF fold%=1 AND FOLDED%=1 THEN 8
20
  700
  710 VDU 26
  720 PROCdraw
  730
  740 draw%=1
  750
  760 PROCvalue(Hand2$)
  770
  780 IF fold% THEN PROCbet ELSE PROC
bet1
  790
  800 PROCbluff
  810
  820 fold% = fold% +1
  830 UNTIL fold%>1
  840 fold%=0
  850 UNTIL FALSE
  860 END
  870
  880 DEFPROCvalue(Z$)
  890
  900 flush%=0
  910 Y$=Z$
  920 value=0
  930
  940 PROCpremanip
  950
  960 PROCflush
  970
  980 N%=FALSE
  990 FOR J%=1 TO 4
 1000 FOR K%=J%+1 TO 5
 1010 IF MID$(Z$,2*J%-1,1)=MID$(Z$,2*
K%-1,1) THEN N%=N%+1:A%(J%,K%)=1
 1020 NEXT
 1030 NEXT
 1040 IF N%=1 THEN PROCa:ENDPROC
 1050 IF N%=2 THEN PROCb:ENDPROC
 1060 IF N%=3 THEN PROCc:ENDPROC
 1070 IF N%=4 THEN PROCd:ENDPROC
 1080 IF N%=6 THEN PROCe:ENDPROC
 1090
 1100 PROCflush
 1110
 1120 REM    flush
 1130
 1140 FOR J%= 1 TO 4
 1150 FOR K%= J%+1 TO 5
 1160 IF MID$(Z$,2*J%,1) = MID$(Z$,2*
K%,1)THEN N%=N%+1:A%(J%,K%)=1
 1170 NEXT
 1180 NEXT
 1190 IF N%=10 THEN flush%=1
 1200
 1210 PROCstraight
 1220 IF value > 0 THEN ENDPROC
 1230
 1240 REM    bust
 1250
 1260 ZZ$=""
 1270 FOR J%=1 TO 9 STEP 2
 1280 IF MID$(Z$,J%,1)="="THEN value
= 13:ZZ$=ZZ$+"K"+MID$(Z$,J%+1,1)
 1290 IF MID$(Z$,J%,1)=">"OR MID$(Z$,
J%,1)="1"THEN value = 14:ZZ$=ZZ$+"A"+
MID$(Z$,J%+1,1)
 1300 NEXT
 1310 IF value<13 AND cheat%=0 THEN F
OLD% = 1
 1320
 1330 ENDPROC
 1340
 1350 DEF PROCstraight
 1360
 1370 FOR J%=1 TO 5:Z$(J%)="":Z$(J%)=
MID$(Z$,2*J%-1,2):NEXT
 1380 FOR K%=1 TO 4
 1390 least%=K%
 1400    FOR test% = K%+1 TO 5
 1410      IF Z$(test%) < Z$(least%)
THEN  least% = test%
 1420    NEXT
 1430
 1440 PROCswap
 1450
 1460 NEXT
 1470 Z$=""
 1480 FOR J%=1 TO 5
 1490 Z$=Z$+Z$(J%)
 1500 NEXT
 1510 ZZ$=Z$
 1520 BB%=0
 1530 AA%=ASC(MID$(Z$,1,1))
 1540 FOR J%=5 TO 2 STEP -1
 1550 BB%=BB%+(ASC(MID$(Z$,2*J%-1,1))
-AA%)
 1560 NEXT
 1570 IF BB%=10 AND flush%=0 THEN val
ue = (AA%-44)*20000
 1580 IF BB%=10 AND flush%=1 THEN val
ue = (((AA%-44)*10)+30000200)
 1590 IF BB%>10 AND flush%=1 THEN val
ue = (ASC(Z$(5))-48)*200000+(ASC(Z$(4
))-48)*10000+(ASC(Z$(3))-48)*500+(ASC
(Z$(2))-48)*25+(ASC(Z$(1))-48)
 1600 IF BB%>10 AND flush%=1 AND ace%
=1 AND run%=FALSE THEN Z$=LEFT$(IT$,p
osition%-1)+"1"+RIGHT$(IT$,10-positio
n%):run%=1:PROCstraight:run%=FALSE
 1610 IF BB%>10 AND flush%=FALSE AND
ace%=1 AND run%=FALSE THEN Z$=LEFT$(I
T$,position%-1)+"1"+RIGHT$(IT$,10-pos
ition%):run%=1:PROCstraight:run%=FALS
E
 1620 ENDPROC
 1630
 1640 DEF PROCpremanip
 1650 FOR J%=1 TO 9 STEP 2
 1660 z$=Z$
 1670 IF MID$(z$,J%,1)="T"THEN Z$=LEF
T$(z$,J%-1)+":"+RIGHT$(z$,10-J%)
 1680 IF MID$(z$,J%,1)="J"THEN Z$=LEF
T$(z$,J%-1)+";"+RIGHT$(z$,10-J%)
 1690 IF MID$(z$,J%,1)="Q"THEN Z$=LEF
T$(z$,J%-1)+"<"+RIGHT$(z$,10-J%)
 1700 IF MID$(z$,J%,1)="K"THEN Z$=LEF
T$(z$,J%-1)+"="+RIGHT$(z$,10-J%)
 1710 IF MID$(z$,J%,1)="A"THEN Z$=LEF
T$(z$,J%-1)+">"+RIGHT$(z$,10-J%):IT$=
Z$:ace%=1:position%=J%
 1720 NEXT
 1730 y$=Z$
 1740 ENDPROC
 1750
 1760 DEF PROCflush
 1770
 1780 FOR J%=1 TO 4
 1790 FOR K%=1 TO 5
 1800 A%(J%,K%)=FALSE
 1810 NEXT
 1820 NEXT
 1830 ENDPROC
 1840
 1850 DEF PROCflush1
 1860 FOR J%=0 TO 14
 1870 type%(J%) = FALSE
 1880 NEXT
 1890 ENDPROC
 1900
 1910 DEF PROCflush2
 1920 FOR J%=1 TO 5:Z%(J%)=FALSE:NEXT
 1930 ENDPROC
 1940
 1950 DEF PROCit
 1960 FOR J%=1 TO 5
```

```
1970 K%=(ASC(MID$(Z$,2*J%-1,1)))-48
1980 type%(K%) = type%(K%) + 1
1990 NEXT
2000 ENDPROC
2010
2020 DEF PROCswap
2030  temp$ = Z$(K%)
2040  Z$(K%) = Z$(least%)
2050  Z$(least%) = temp$
2060 ENDPROC
2070
2080 DEF PROCa
2090
2100 PROCflush2
2110
2120 FOR J%=1 TO 4
2130 FOR K%=1 TO 5
2140 IF A%(J%,K%)=1 THEN C%=J%:D%=K%

 2150 NEXT
 2160 NEXT
 2170 ZZ$=MID$(Y$,2*C%-1,2)+MID$(Y$,2
*D%-1,2)
 2180 Z%(C%)=1:Z%(D%)=1:YY$=""
 2190 FOR J%= 1 TO 5
 2200 IF Z%(J%)=FALSE THEN YY$=YY$+MI
D$(y$,2*J%-1,1)
 2210 NEXT
 2220 FOR J%=1 TO 3
 2230 YY$(J%)=MID$(YY$,J%,1)
 2240 NEXT
 2250 IF YY$(1)<YY$(2) THEN T$=YY$(2)
:YY$(2)=YY$(1):YY$(1)=T$
 2260 IF YY$(2)<YY$(3) THEN T$=YY$(3)
:YY$(3)=YY$(2):YY$(2)=T$
 2270 IF YY$(1)<YY$(2) THEN T$=YY$(2)
:YY$(2)=YY$(1):YY$(1)=T$
 2280 value = (ASC(MID$(z$,2*C%-1,1))
-48)*100+(ASC(YY$(1))-48)+(ASC(YY$(2)
)-48)*.01+(ASC(YY$(3))-48)*.0001
 2290 ENDPROC
 2300
 2310 DEF PROCb
 2320
 2330 PROCflush2
 2340
 2350 ZZ$="":ABC%=FALSE
 2360 FOR J%=1 TO 4
 2370 FOR K%=1 TO 5
 2380 IF A%(J%,K%)=1 THEN ZZ$=ZZ$+MID
$(Y$,2*J%-1,2)+MID$(Y$,2*K%-1,2):Z%(J
%)=1:Z%(K%)=1:ABC%=ABC%+1:local%(ABC%
)=ASC(MID$(Z$,2*J%-1))-48
 2390 NEXT
 2400 NEXT
 2410 FOR J%=1 TO 5
 2420 IF Z%(J%)=FALSE THEN local% = A
SC(MID$(Z$,2*J%-1,1))-48
 2430 NEXT
 2440 IF local%(1)>local%(2) THEN val
ue = local%(1)*500+local%(2)*20+local
% ELSE value = local%(2)*500+local%(1
)*20+local%
 2450 ENDPROC
 2460
 2470 DEF PROCc
 2480 ZZ$=""
 2490 FOR J%=1 TO 5
 2500 Z%(J%)=FALSE
 2510 NEXT
 2520 FOR J%=1 TO 4
 2530 FOR K%=1 TO 5
 2540 IF A%(J%,K%)=1 THEN Z%(J%)=1:Z%
(K%)=1
 2550 NEXT
 2560 NEXT
 2570 FOR J%=1 TO 5
 2580 IF Z%(J%)=1 THEN ZZ$=ZZ$+MID$(Y
$,2*J%-1,2):CBA%=J%
 2590 NEXT
 2600 value = (ASC(MID$(Z$,2*CBA%-1,1
))-48)*5000
 2610 ENDPROC
 2620
 2630 DEF PROCd
 2640
 2650 PROCflush1
 2660
 2670 PROCit
 2680
 2690 FOR J%=1 TO 14
 2700 IF type%(J%) = 3 THEN value = J
%*2E6
 2710 NEXT
 2720 ZZ$=Y$
 2730
 2740 ENDPROC
 2750
 2760 DEF PROCe
 2770
 2780 PROCflush1
 2790
 2800 PROCit
 2810
 2820 FOR J%=1 TO 14
 2830 IF type%(J%) = 4 THEN value = (
J%*10)+3E7
 2840 NEXT
 2850 ZZ$=Y$
 2860
 2870 ENDPROC
 2880
 2890 DEF PROCdeal(HAND$,Y%)
 2900 blip% = 1
 2910 FOR X%=100 TO 900 STEP 200
 2920 K$=MID$(HAND$,blip%,2)
 2930 PROCcard(X%,Y%)
 2940 PROCval
 2950 blip% = blip% + 2
 2960 NEXT X%
 2970 VDU 4
 2980 ENDPROC
 2990
 3000 DEF PROCcard(X%,Y%)
 3010 GCOL 0,3
 3020 MOVE X%,Y%
 3030 MOVE X%+160,Y%
 3040 PLOT 85,X%,Y%+200
 3050 PLOT 85,X%+160,Y%+200
 3060 GCOL 0,0
 3070 ENDPROC
 3080
 3090 DEF PROCval
 3100 X$=MID$(K$,1,1)
 3110 VDU 5
 3120 IF X$="T" THEN MOVE X%+24,Y%+11
2:PRINT"10":GOTO3140
 3130 MOVE X%+8,Y%+180:PRINT X$:MOVE
X%+104,Y%+40:PRINT X$
 3140 X$=MID$(K$,2,1)
 3150 IF X$="C" THEN N%=1
 3160 IF X$="D" THEN N%=2
 3170 IF X$="H" THEN N%=3
 3180 IF X$="S" THEN N%=4
 3190 MOVE X%+88,Y%+180:PRINT ;ST$(N%
):MOVE X%+8,Y%+40:PRINT ;ST$(N%)
 3200 ENDPROC
 3210
 3220 DEF PROCinfo
 3230 VDU28,7,30,18,29
 3240 COLOUR 131
 3250 CLS
 3260 COLOUR 0
 3270 PRINT"pot    ";POT%
 3280 PRINT"credit ";
 3290 IF STAKE% < 0 THEN COLOUR 1
 3300 PRINT;STAKE%;
 3310 COLOUR 0
 3320 COLOUR 130
 3330 VDU 26
 3340 ENDPROC
 3350
 3360 REM I fold
 3370
 3380 *FX15
 3390 IF value < 500 AND cheat%=FALSE
 THEN PRINT TAB(0,25);"I fold":O$=INK
EY$(500):GOTO 820
 3400 PRINT TAB(0,25);"I am not going
 to  fold"
 3410 O$=INKEY$(500)
 3420 PRINT TAB(0,25);SPC(30)
 3430 GOTO 630
 3440
 3450 REM you fold
 3460
 3470 PRINT TAB(0,25);"Do you want to
 fold"
 3480 REPEAT:O$=GET$:UNTIL O$="Y" OR
O$="N"OR O$="y"OR O$="n"
 3490 IF O$="Y"OR O$="y" THEN 820
 3500 PRINT TAB(0,25);SPC(30)
 3510 GOTO 630
 3520
```

→

SATAN'S CHALLENGE

or (Nevil Rides Out) A Black Magic Adventure

Dare you take up the challenge laid down by the most evil and sinister of all beings . . . the devil himself.

If you do you will find yourself alone and at the mercy of the twisted fancies and whims of a cold and calculating mind.

Occult forces are threatening the lives of those near and dear to you. Their only salvation rests in your hands but in accepting the challenge your own existence is put in severe peril.

In taking up the challenge you have to find the TALISMAN and locate a pentagram which then has to be prepared for the final rite. In the mean time dark forces will be opposing you making a difficult task almost impossible.

Do not allow yourself to the lulled into a sense of security for it will be short lived.

Many have gone before only to swell the ranks of the damned.

This is the latest adventure from the stables of Microtest and has been written with the acclaimed features of other adventures in mind eg save facility, quick response, simple but extensive commands, a mixture of logical and fiendish problems to solve.

Be warned this is an easy adventure to get into but devilishly difficult to end.

*CASSETTE **£7.95** inc VAT*

*DISC 40 or 80 tk **£9.45** inc VAT (add 50p p&p)*

DAIRYFILE FOR DAIRY FARMERS

Keep on that economic line between over and under feeding!

Save time recording milk yield and calculating feed amounts!

Quickly decide feeding policy with the 'Monthly Calving Group' Performance Graph!

Print out a recording sheet with cows in numerical order. Print out graphs or tables of individual cows or Monthly Groups showing serving and pregnancy details, illness record, Calving Index, weekly and running total Margin over Concentrate. See at a glance cows due for serving, pregnancy diagnosis and drying off. Keep track of weekly total feed cost and milk sale price.

All this and more with DAIRYFILE.

Predict cow or Monthly Group total lactation yield. Compare with Standard Lactation Curves. All old data preserved – compare Monthly Groups over the years. Which is the most profitable month?

Find out with DAIRYFILE – for up to 200 lactating cows.

Available on dual 40 tk disc (For dual 40 tk drives only)

Dual or Single 80 tk drives **£69** inc VAT (add 50p p&p)

Please state on ordering which disc format and printer model.

MICROTEST FONT ROM

This exciting new ROM from Microtest will enable you to get all sorts of new characters and fonts from your BBC Computer. Once you have produced your masterpiece on the screen, all you have to do is use the inbuilt screen-dump utility to produce a hard copy onto paper.

Typing '*HELP FONTS' gives a list of available fonts and the blocks of characters which they replace.

Available fonts are:

***Accents**	Accents and miscellaneous.
***Block**	Small capitals.
***Data**	Like the bottoms of cheques.
***Greek**	It's all Greek to me too!
***Joined**	Standard capitals with joined up lower case.
***Maths**	A mix of until now unobtainable Mathematical symbols.
***Miscellaneous**	A few oddities which often are very necessary.
***Thick**	Thick text (for MODEs 0&3) to enhance 80 column mode.
***Thin**	Thin text (for MODEs 2&5) which makes modes 2 & 5 much more readable or perhaps "READABLE".
***Vertical**	For labelling graphs.
***MODE 8**	Ten column multicolour, memory-miserly mode.
***S DUMP**	A dump facility which will produce a screen dump of any MODE from 0 to 8 (including a text only dump in MODE 7) on an Epson, Star, CTI CP80 or MT80.

The ROM uses absolutely NO user memory and can be used with word processors etc. as well as normal BASIC programs. The DUMP utility will produce positive or negative graphics and will also rotate the picture produced through 90° as well as being able to position the picture anywhere laterally across the sheet of paper.

***£17.50** inc VAT (add 50p p&p)*

Microtest Starstick ROM & Joystick Package

Now available the Starstick ROM and Joystick. This comes in three forms:

(A) The Starstick ROM and Quickshot I Joystick
Price 17.95 + VAT = 19.84

(B) The Starstick ROM and Quickshot II Joystick
Price 19.25 + VAT = 22.14

(C) The Starstick ROM and patch lead, choose your own Spectrum/Atari style joystick
Price 15.25 + VAT = 17.54

Post and Packing £1.00 inc. VAT per item

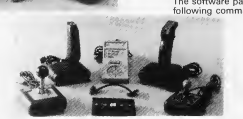
This enables you to plug the Rapid action self centring joysticks until now only available for the Spectrum/Atari/CBM machines into the user port of the BBC. Model A users please note NO ANALOGUE INTERFACE REQUIRED.

Disc Users Note – pressing BREAK, SHIFT-BREAK or CONTROL BREAK does not modify or destroy the STARSTICK software so Disc Users please feel free to Boot!

The software patch provided in the ROM is interrupt driven and adds the following commands to your computer.

★ STICK turn on the STARSTICK ROM
★ NSTICK turn off the STARSTICK ROM
★ SETSTICK set up joystick to users spec
★ SAVE "NAME" 140 160 saves your user-key protocols
★ ADVAL emulate standard analogue joysticks
★ PAUSE define key to Freeze game
★ NPAUSE turn off ability to freeze game
★ "NAME" predefined key protocols set up for software houses programs
★ HELP KEYS displays currently selected key protocols
★ REPEAT enables auto-repeat fire
★ NREPEAT disables auto-repeat fire

Enables you to use our joysticks on most programs even those that do not offer joystick capability.



DEALER ENQUIRIES
and
EXPORT ORDERS
WELCOME

available from

MICROTEST LTD
18 Normandy Way, Bodmin, Cornwall PL31 1EX
Telephone: 0208 3812
OR ANY GOOD DEALER

VISA

Access

Large picture shows BBC Computer System and a Quickshot II Joystick. Small inset just a few of the joysticks that will work with the patch lead. Screenshot by kind permission of SUPERIOR SOFTWARE

All MICROTEST ROMs are fully TUBE® compatible. (TUBE ® is a reg. trademark of Acorn Computers).

From Page 17

```
 3530 DEF PRDCbet
 3540
 3550 1F cheat% THEN PRDCcheat ELSE 1
F draw% THEN PRDCpre_bet1 ELSE PRDCpr
e_bet
 3560
 3570 VDU 28,1,27,18,22
 3580 COLOUR 128:CLS:COLDUR3
 3590
 3600 1F FOLD% = 1 AND cheat%=FALSE T
HEN PRINT "I fold":STAKE%=STAKE%+POT%
:POT%=FALSE:PROCinfo:D$=INKEY$(500):E
NDPROC
 3610 PR1NT"I open with ";:BET%=bet%+
1NT(RND(bet%)):PR1NT;BET%
 3620 POT%=POT%+BET%:PROCinfo
 3630 PROC2nd_bit:ENDPROC
 3640
 3650 DEF PRDC2nd_bit
 3660 VDU 28,1,27,18,24
 3670 COLOUR3:COLOUR128
 3680 PRINT"1. See"
 3690 PR1NT"2. Raise"
 3700 PRINT"3. Fold"
 3710 REPEAT:opt%=GET:UNT1L opt%>48 A
ND opt%<52
 3720 opt%=opt%-48
 3730 IF opt%=1 AND draw% = 1 THEN ST
AKE%=STAKE%-BET%:PDT%=POT%+BET%:COLOU
R130:CLS:VDU26:PROCinfo:PRDCwin:ENDPR
OC
 3740 1F opt%=1 THEN STAKE%=STAKE%-BE
T%:POT%=POT%+BET%:PROCinfo:ENDPROC
 3750 IF opt% = 3 THEN FOLDED% = 1:PO
T%=0:COLOUR 130:COLOUR 0:VDU26:CLS:EN
DPROC
 3760 REPEAT
 3770 CLS:1NPUT"How euch",raise1%
 3780 UNT1Lraise1%>0 AND raise1%<=10
 3790 BET1%=BET%+raise1%:STAKE%=STAKE
%-BET1%:POT%=POT%+BET1%
 3800 1F BET1% <= (raise%*bet%) OR bl
uff% > 25 DR cheat%=1 DR POT%>20 PR1N
T"1 see you":POT%=POT% + BET1%:BLOP%=
1
 3810 1F BLDP%=1 AND draw% = FALSE TH
EN PROCinfo:D$=1NKEY$(500):BLOP%=FALS
E:ENDPROC
 3820 1F BLOP%=1 AND draw% = 1 THEN P
ROCinfo:O$=1NKEY$(500):PROCwin:BLOP%=
FALSE:ENDPROC
 3830 STAKE%=STAKE%+POT%:POT%=FALSE:F
OLD%=1
 3840 PR1NT"I fold"
 3850 PROCinfo
 3860 O$=INKEY$(500)
 3870 ENDPROC
 3880
 3890 DEF PROCpre_bet
 3900
 3910 IF value < 13 AND draw% = FALSE
 THEN FOLD% = 1:ENDPROC
 3920 IF value < 300 AND draw% = 1 TH
EN FOLD% = 1:ENDPROC
 3930 IF value < 500 THEN bet% = 1: r
aise%= 1:ENDPROC
 3940 1F value < 1000 THEN bet% = 1:
raise%= 2:ENDPROC
 3950 1F value < 1500 THEN bet%= 2: r
aise%= 2:ENDPROC
 3960 IF value < 4000 THEN bet% = 3:
raise%= 3:ENDPROC
 3970 1F value < 8000 THEN bet% = 3:
raise%= 4:ENDPRDC
 3980 1F value < 40000 THEN bet% = 4:
 raise%= 5:ENDPROC
 3990 1F value < 70000 THEN bet% = 4:
 raise%= 6:ENDPROC
 4000 bet% = 5:raise%= 7:ENDPRDC
 4010
 4020 DEF PROCbluff
 4030
 4040 PROCvalue(Hand1$)
 4050
 4060 IF value < 500 THEN BLUFF%=BLUF
F%+1
 4070 bluff% = BLUFF% * 100 D1V no%
 4080 ENDPROC
 4090
 4100 DEF PROCwin
 4110
 4120 VDU 28,1,27,18,22
 4130 COLOUR130:COLOUR0:CLS
 4140 VDU26
 4150
 4160 PRDCvalue(Hand1$)
 4170 value1=value
 4180
 4190 PROCvalue(Hand2$)
 4200 value2=value
 4210
 4220 PRDCdeal(Hand2$,400)
 4230
 4240 1F value1>value2 THEN PR1NT TAB
(0,21);"you win ";Dpponent$:STAKE%=ST
AKE%+PDT%:POT%=0:PROCinfo:O$=1NKEY$(5
00):ENDPRDC
 4250 IF value1<value2 THEN PR1NT TAB
(0,21);"you lose ";Opponent$:POT%=0:P
ROCinfo:O$=INKEY$(500):ENDPROC
 4260 1F value1=value2 THEN PRINT TAB
(0,21);"both hands are of equal value
 ";Opponent$:STAKE%=STAKE%+(POT%/2):P
OT%=0:PROCinfo:O$=1NKEY$(500):ENDPROC
 4270
 4280 DEF PROCdraw
 4290 VDU 26
 4300 VDU 28,1,27,18,22:CDLDUR128:CDL
DUR3
 4310 CLS
 4320 len%=LEN(ZZ$)
 4330 1F len% = 10 THEN PRINT TAB(0,2
2);"I take no cards" ELSE 1F (10-len%
)/2=1 THEN PR1NT TAB(0,22);"1 take 1
card" ELSE PRINT TAB(0,22);"1 take ";
(10-len%)/2;" cards"
 4340 IF len%=10 THEN 4410
 4350 a%=(10-len%)/2+Count%
 4360 REPEAT
 4370 ZZ$=ZZ$+A$(Count%)
 4380 Count%=Count%+1
 4390 UNT1L Count% = a%
 4400 Hand2$=ZZ$
 4410 PRINT"Do you want to    change
any cards"
 4420 REPEAT:O$=GET$:UNT1L O$="Y" OR
O$="N"OR O$="y"OR O$="n"
 4430 IF D$="N"DR O$="n" THEN COLOUR1
30:COLOUR0:CLS:VDU26:ENDPROC
 4440
 4450 PRDCflush2
 4460
 4470 FDR J%=1 TO 5
 4480 PR1NT"Change card ";J%;" ?";
 4490 REPEAT:O$=GET$:UNTIL O$="Y" OR
O$="N"OR D$="n"OR D$="y"
 4500 PR1NTO$
 4510 1F O$="Y"DR O$="y" THEN Z%(J%)=
1:taken%=taken%+1
 4520 NEXT
 4530 len%=0:O$=""
 4540 FOR J%=1 TO 5
 4550 1F Z%(J%)=0 THEN O$=O$+M1D$(Han
d1$,2*J%-1,2)
 4560 IF Z%(J%)=1 THEN len%=len%+1
 4570 NEXT
 4580 1F LEN(O$)>9 THEN 4640
 4590 a%=len%+Count%
 4600 REPEAT
 4610 O$=O$+A$(Count%)
 4620 Count%=Count%+1
 4630 UNT1L a%=Count%
 4640 Hand1$=O$
 4650 PROCdeal(Hand1$,700)
 4660 COLOUR130:CDLDUR0:CLS:VDU26:END
PRDC
 4670
 4680 DEF PRDCcheat
 4690
 4700 FOLD%=0
 4710
 4720 PROCvalue(Hand1$)
 4730 value1=value
 4740
 4750 PRDCvalue(Hand2$)
 4760 value2=value
 4770
 4780 1F value2>value1 THEN bet%=5:ra
ise%=7:ENDPROC
 4790
 4800 PROCalter
 4810 PROCvalue(Hand2$)
```

SOFT SHOP

The best software store in the country
(now boasting over 2,500 good titles in stock)
with accessories, joysticks and books. We have
somehow made room for hardware stocks.
Home computers, disc drives, printers, monitors,

and now a very special offer

SONY®

R.G.B. MONITOR*

Ideal for BBC/Electron and many others.

- ✱ 14″ colour monitors
- ✱ Resolution equivalent to 480 x 400 pixels
- ✱ Full 12 month guarantee

✱ These monitors using
Trinitron tubes have
been converted to RGB

Our unbeatable price **£149.50** inc. VAT.

Or mail order £159.50 including VAT, p&p & BBC leads

Postal and telephone orders welcomed.
All major credit cards accepted.
(Offer subject to availability)

Easy Road Access
Free Parking and
Open Mon. — Sat.
9.00–5.30
Sun. 9.30 - 12.30

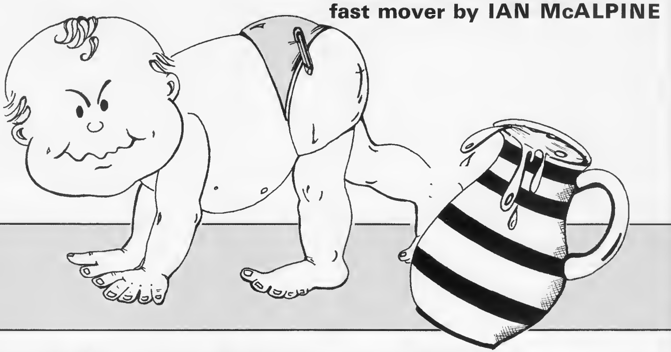
Soft Shop Ltd.
55 St Peter's Court
Chalfont St Peter
Bucks. SL9 9QQ

Tel: (0753) 889010

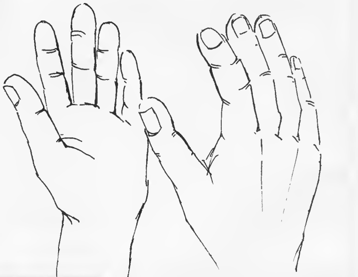
From Page 19

```
 4820 IF value<10000 THEN PROCalter1
 4830 ENDPROC
 4840
 4850 DEF PROCalter
 4860
 4870 abo%=abo%+1
 4880 IF abo%=2 THEN ENDPROC
 4890 S$=MID$(Hand2$,1,2)
 4900 naughty%=16
 4910 REPEAT
 4920 F$=A$(naughty%)
 4930 IF MID$(S$,1,1)=MID$(F$,1,1) TH
EN S$=S$+F$
 4940 naughty%=naughty%+1
 4950 UNTIL naughty%=53
 4960 abbo%=LEN(S$)
 4970 abbo$=Hand2$
 4980 Hand2$=S$+MID$(abbo$,3,10-abbo%
)
 4990 ZZ$=Hand2$
 5000 bet%=5:raise%=7
 5010
 5020 ENDPROC
 5030
 5040 DEF PROCbet1
 5050
 5060 IF cheat% THEN PROCcheat ELSE I
F draw% THEN PROCpre_bet1 ELSE PROCpr
e_bet
 5070
 5080 VDU28,1,27,18,22
 5090 COLOUR128:COLOUR3:CLS
 5100 PRINT"Do you want to:"
 5110 PRINT"1. Open"
 5120 PRINT"2. Fold"
 5130 REPEAT:opt%=GET:UNTIL opt%>48 A
ND opt%<51
 5140 IF opt%=50 THEN FOLDED%=1:COLOU
R130:COLOUR0:VDU26:CLS:POT%=0:ENDPROC

 5150 REPEAT
 5160 PRINTTAB(0,4);SPC(14)
 5170 INPUT TAB(0,4)"How much",BET%
 5180 UNTIL BET%>0 AND BET%<=10
 5190 *FX15
 5200 STAKE%=STAKE%-BET%:POT%=POT%+BE
T%
 5210 IF ((bet%*raise%)>=BET% OR bluf
f%>25 OR POT%>20 OR cheat%=1) AND FOL
D%=0 THEN CLS:PRINT"I see you":POT%=P
OT%+BET%:PROCinfo:BLOP%=1 ELSE FOLD%=
1
 5220 IF FOLD%=1 AND cheat%=FALSE THE
N PRINT"I fold":STAKE%=STAKE%+POT%:PO
T%=0:PROCinfo:VDU26:O$=INKEY$(500):EN
DPROC
 5230 g=(raise%*bet%-BET%)/2
 5240 IF g>10 THEN g=10
 5250 IF g>1 THEN g=INT(g):COLOUR128:
COLOUR3:PRINTTAB(1,23)"and raise you
";g;:POT%=POT%+g:BET%=g:PROCinfo:PROC
2nd_bit
 5260 IF draw% AND g>=1 AND FOLDED%=0
PROCwin:VDU26
 5270 IF FOLDED%=0 O$=INKEY$(500)
 5280 ENDPROC
 5290
 5300 DEF PROCpre_bet1
 5310
 5320 IF value<13 AND draw%=0 THEN FO
LD%=1:ENDPROC
 5330 IF value<300 AND draw%=1 THEN F
OLD%=1:ENDPROC
 5340 IF value<500 THEN bet%=1:raise%
=1:ENDPROC
 5350 IF value<1000 THEN bet%=1:raise
%=2:ENDPROC
 5360 IF value<1200 THEN bet%=2:raise
%=2:ENDPROC
 5370 IF value<1500 AND taken%>2 THEN
 bet%=4:raise%=2:ENDPROC ELSE IF valu
e<1500 AND taken%<3 THEN bet%=2:raise
%=2:ENDPROC
 5380 IF value<4000 AND taken%>2 THEN
 bet%=5:raise%=3:ENDPROC ELSE IF valu
e<4000 AND taken%<3 THEN bet%=3:raise
%=3:ENDPROC
 5390 IF value<8000 AND taken%>2 THEN
 bet%=5:raise%=5:ENDPROC ELSE IF valu
e<8000 AND taken% <3 THEN bet%=2:rais
e%=3:ENDPROC
 5400 IF value<40000 AND taken%>2 THE
N bet%=5:raise%=5:ENDPROC ELSE IF val
ue<40000 AND taken% <3 THEN bet%=4:ra
ise%=4:ENDPROC
 5410 IF value<70000 AND taken%>2 THE
N bet%=5:raise%=5:ENDPROC ELSE IF val
ue<70000 AND taken% <3THENbet%=4:ra
ise%=4:ENDPROC
 5420 bet%=5:raise%=7:ENDPROC
 5430
 5440 DEF PROCalter1
 5450
 5460 stiff%=stiff%+1
 5470 IF stiff%=2 THEN ENDPROC
 5480 naughty%=16
 5490 F$=MID$(Hand2$,1,2)
 5500 REPEAT
 5510 naughty%=naughty%+1
 5520 O$=A$(naughty%)
 5530 IF MID$(F$,2,1)=MID$(O$,2,1) TH
EN F$=F$+O$
 5540 UNTIL naughty%=52 OR LEN F$=10
 5550 IF LEN F$<10 THEN Hand2$=F+RIG
HT$(F$,10-LEN(F$        ):ENDPROC
 5560 Hand2$=F$
 5570 ZZ$=Hand2$
 5580 ENDPROC
 5590
 5600 REM these DATA statements shoul
d            be filled with names of
people          you want the computer
 to cheat           against.
 5610
 5620 DATA ****
 5630
 5640 REM Escape Key has been Pressed
 5650 MODE7
 5660 END
```

Catch crashing crockery in this fast mover by IAN McALPINE



BABY, YOU'RE QUITE A HANDFUL!



YOUR rich aunt and uncle have asked you to babysit with their two-year-old son, Tobermory, who is quite a mischievous chap for his age.

No sooner have your aunt and uncle left than he jumps up on to the china cabinet and runs along the top of it, jumping up and down as he goes.

The expensive china starts to fall and you run frantically left and right trying to catch the falling pieces.

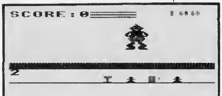
It's easy to start with, but soon the crockery falls thick and fast.

Because of the Law of Frustrating Games the china will occasionally turn into a red anti-matter blob, heralded by a ping.

Never try to catch an anti-matter blob or your hands will explode and you will lose a life – as you do when you drop a piece of china.

You have three lives to begin with, but get an extra one on completion of Level 3. The number of lives left is printed in yellow two-thirds up the left hand side of the screen. This number turns red when you only have one life left.

Once you have lost all of your lives, your aunt and uncle return home, see their beautiful china smashed, and uncle promptly shoots you!



SCORE : 0

2

The program's features include ability to turn the sound on/off, use of keyboard or joystick for movement, pause game control, SAVE and LOAD high score from cassette, "gruesome consequences" and full use of the Acorn Speech System if fitted.

The Speech System is used for prompting input, speaking level number, the "thud" in the gruesome consequences and for announcing the score.

Enter the program exactly as listed, as it runs very tight on memory.

When entering the program disc users would be advised to SAVE it each time before they run it because of the relocator at the start of the program.

Cassette users can leave out lines 60, 1330 and 1340. All users should initially enter line 70 as:

```
70 PROCi
```

Once the program is running correctly the rest of line 70 can be entered.

CONTROLS

- *To move your hands left or right use the left or right cursor keys.*
- *To accelerate hold down either Shift key.*
- *To pause the game press Shift Lock.*
- *To continue press Caps Lock.*
- *To turn the sound on/off press S/Q. The sound controls only take effect at the end of each drop of china.*
- *To return to the title page press Escape.*

PROCEDURES

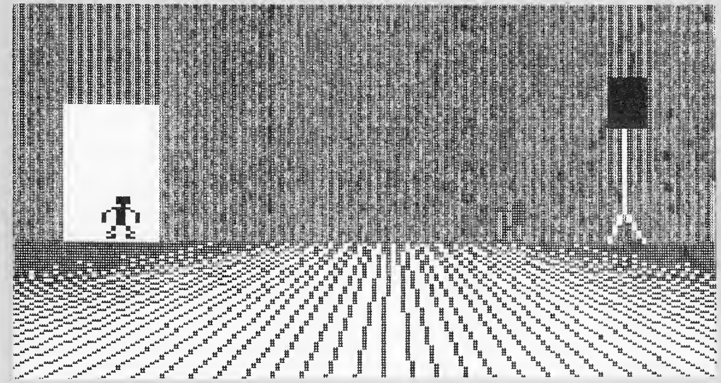
PROCi	Initialises characters and envelopes.
PROCgh	Asks whether to load recorded high score.
PROChp	Prints title page.
PROCa	Presses space or fire button.
PROCi1	Initialises strings and secondary variables.
PROCre	Prepares to start game.
PROCw	Checks whether using keys or joystick.
PROCb	Prints bouncing baby.
PROCf	Drops a piece of china.
PROCwt(num)	Delay.
PROCu	Updates score, high score and pieces of china left to drop.
PROCff	Plays title tune.
PROCc	Gruesome consequences.
PROCp	Prints your score and high score at end of game.
PROCs	Prints screen display.
PROCkm	Checks which keys pressed.
PROCjm	Checks for joystick movement.
PROCpa	Pauses game or sound on/off?
PROCct	You caught the piece of china.
PROCdp	You dropped the piece of china.
PROCe	Erases piece of china from bottom of screen.
PROCrh	Records high score.
PROCr	Restores everything to default.
PROCas	Assembles machine code to print the baby and your hands.
PROCi2	Initialises which piece of china goes where on the cabinet.
PROCspk	Reads data, then speak.
PROCcd (num)	Prints "China Drop", in teletext graphics.
PROCdo (num)	Plot door.
FNrc (num)	Reads character.

```
   10 REM ...China Drop...
   20 REM by Ian McAlpine.
   30 REM Theme, Graphics and Speech
   40 REM consultant, Keith McAlpine.
   50 REM (C) The Micro User
   60 IFPAGE<>&E00THEN1330
   70 PROCi:ONERRORMODE7:CLEAR:VDU23;
10,32,0;0;0;:PROChp:PROCa:GOTO90
   80 MODE7:VDU23;10,32,0;0;0;:PROCgh
:CLS:PROChp:PROCa
   90 PROCi1:PROCas:MODE2:PROCre:SOUN
D3,5,50,-1:REPEAT:PROCw:PROCb:PROCf:I
Fd=8ANDcr<>0sk=sk+1:1I=Il+1:IFsk=5cr=
cr+1:PROCwt(50):SOUND1,4,200,5:PROCu:
PROCwt(150):PROCff:PROCwt(150)
  100 IFsk>=5sk=5
  110 IFd=8ANDcr<>0MODE2:PROCre:SOUND
3,5,50,-1
  120 UNTILcr=0:*FX15,0
  130 CLS:PROCc:MODE7:VDU23;10,32,0;0
;0;:PROCp:END
  140 DEFPROCi:RESTORE1360:FORc%=223T
O254:VDU23,c%:FORd%=0TO7:READd:VDUd:N
EXT,:ENVELOPE1,4,-4,-1,-1,20,20,20,1,
0,0,0,1,1:ENVELOPE2,133,8,4,8,3,1,1,1
26,0,0,-10,126,0:ENVELOPE3,136,-1,-1,
-1,30,2,2,127,0,-127,-10,60,60
  150 ENVELOPE4,1,0,0,0,0,0,0,126,-1,
0,-3,126,126:ENVELOPE5,1,40,-4,40,6,1
2,6,50,0,0,-50,75,5:H%=100:*FX4,1
  160 ENDPROC
  170 DEFPROCi1:DIMO$(9),C%9:*FX11,1
  180 *FX12,100
  190 sc=0:cr=3:sk=2:Il=1:pp=3:RESTOR
E1410:FORD%=&0A40TO&0A84:READD:?D%=D:
NEXT:ENDPROC
  200 DEFPROCi2:FORch=1TO8:O$(ch)=CHR
$(244+RND(4)):C%?ch=RND(7):NEXT:O$(9)
=CHR$244:C%?9=1:xh=8:yh=30:xb=8:yb=2:
wa=2:d=0:y=FALSE:rep=FALSE:ENDPROC
  210 DEFPROCa:PROCff:PROCwt(150):VDU
31,0,23,129,157,135,136:PRINT"PRESS <
RETURN> OR 'FIRE' BUTTON ..";:RESTORE
1420:PROCspk:TIME=0:REPEAT:k=INKEY(-7
4):j=ADVAL(0):UNTILk=TRUE ORj=1ORTIME
>=1500:IFk=TRUE a$="K"ELSEa$="J"
  220 IFTIME>=1500PROCa
  230 ENDPROC
  240 DEFPROCs:VDU19,7,0;0;19,1,0;0;1
9,3,0;0;:GCOL0,7:MOVE0,736:DRAW1279,7
36:MOVE0,670:DRAW1279,670:MOVE0,30:DR
AW1279,30:MOVE400,992:DRAW700,992:MOV
E400,1007:DRAW700,1007:MOVE400,1019:D
RAW700,1019
  250 GCOL44,1:FORyc=740TO764STEP4:PL
OT77,640,yc:NEXT:GCOL43,3:FORyc=0TO26
STEP4:PLOT77,640,yc:NEXT:PRINTTAB(6,1
5)SPC(7):VDU20:FORpr=1TO8:COLOURC%?pr
:VDU31,pr*2,10:PRINTO$(pr):NEXT:*FX9,
5
```

→

From Page 23

```
  260 *FX10,5
  270 COLOUR14:PRINTTAB(0,0);"SCORE:"
:COLOUR12:PRINTTAB(11,0);"HIGH: ":COLO
UR4:PRINTTAB(6,31);"LEVEL  ";:COLOUR2
:PRINTTAB(6,0);sc:COLOUR5:PRINTTAB(16
,0);H%:COLOURpp:PRINTTAB(0,9);cr:COLO
UR4:PRINTTAB(13,31);lI;
  280 DEFPROCw:IFa$="K"PROCka ELSEPRO
Cja
  290 ENDPROC
  300 DEFPROCka:IFINKEY(-26)AND(xh>0)
xh=xh-1
  310 IFINKEY(-122)AND(xh<15)xh=xh+1
  320 ?&0A43=xh:CALL&A05:IFrep=TRUE r
ep=FALSE:ENDPROC
  330 IFINKEY(-1)rep=TRUE:PROCka
  340 ENDPROC
  350 DEFPROCja:ov=32750:nv=ADVAL(1):
IFnv<(ov-30000)AND(xh<15)xh=xh+1ELSEI
Fnv>(ov+30000)AND(xh>0)xh=xh-1
  360 ?&0A43=xh:CALL&A05:IFrep=TRUE r
ep=FALSE:ENDPROC
  370 fi=ADVAL(0)AND3:IFfi=1rep=TRUE:
PROCja
  380 ENDPROC
  390 DEFPROCspk:REPEAT:READs:SOUND-1
,s,0,0:UNTILs=1:ENDPROC
  400 DEFPROCb:PROCpa:IFyb=2yb=1ELSEy
b=2
  410 ONwa GOTO430,420
  420 IF(xb<15)xb=xb+1:GOTO450
  430 IF(xb>0)xb=xb-1
  440 IFxb=0wa=2
  450 IFxb=15wa=1
  460 ?&0A4D=xb:?&0A4E=yb:CALL&A20:EN
DPROC
  470 DEFPROCwt(de):TIME=0:REPEATUNTI
LTIME>de:ENDPROC
  480 DEFPROCf:f_ch=RND(5):IFf_ch<3EN
DPROC
  490 o=RND(8):PROCw:PROCb:IFO$(o)=""
THEN490
  500 of$=CHR$237+CHR$8+STRING$(sk,CH
R$10)+O$(o):O$(o)="":xo=o*2:b=FALSE:I
Fsk=3bo=27ELSEbo=28
  510 FORdp=10TObo STEPsk:IFb=FALSE:I
FRND(50)<3ANDdp<=23of$=CHR$237+CHR$8+
STRING$(sk,CHR$10)+O$(9):O$(o)="":o=9
:b=TRUE:SOUND2,-15,200,1
  520 PROCw:PROCb:COLOURC%?o:PRINTTAB
(xo,dp);of$:PROCb:PROCw:NEXT:IFo<>9AN
D FNrc(xo)=130ORo<>9AND FNrc(xo)=129P
ROCct ELSEIFo<>9AND FNrc(xo)<>130ORo<
>9AND FNrc(xo)<>129PROCdp
  530 IFo=9AND FNrc(xo)=130ORo=9AND F
Nrc(xo)=129y=TRUE:C%?o=7:PROCdp:C%?o=
1 ELSEIFo=9AND FNrc(xo)<>130ORo=9AND
FNrc(xo)<>129PROCe:PROCct
  540 IFxb<16COLOUR7:VDU31,xb+1,yb,23
8,239,240:PROCwt(60):VDU8,8,8,237,237
,237
  550 d=d+1:IFd<>8SOUND3,5,50,-1:ENDP
ROC
  560 DEFPROCu:COLOUR2:PRINTTAB(6,0);
sc:COLOUR5:PRINTTAB(16,0);H%;:IFcr<=1
pp=1ELSEpp=3
  570 COLOURpp:PRINTTAB(0,9);cr:ENDPR
OC
  580 DEFFNrc(x):LOCALA%,c%:VDU31,x,y
h:A%=135:=(USR(&FFF4)AND&FFFF)DIV&100
  590 DEFPROCct:*FX15,0
  600 SOUND2,2,100,10:IFsk=3PRINTTAB(
xo,28);" "
  610 IFo<>9sc=sc+sk*10ELSEsc=sc+sk*5
  620 IFsc>H%H%=sc
  630 PROCu:ENDPROC
  640 DEFPROCdp:*FX15,0
  650 PROCe:cr=cr-1:PROCu:ye=0:VDU19,
0,7;0;:PROCwt(5):SOUND1,1,100,1:SOUND
0,-15,7,30:VDU19,0,0;0;:VDU29,(xo*64)
+32;32;
  660 FORex=1TO350STEP10:MOVE0,0:GCOL
0,C%?o:PLOT65,0,ye:MOVE0,0:PLOT65,-ex
,32:MOVE0,0:PLOT65,-ex,ye:MOVE0,0:PLO
T65,ex,32:MOVE0,0:PLOT65,ex,ye:MOVE0,
0:GCOL0,0
  670 PLOT65,-ex,ye:MOVE0,0:PLOT65,ex
,ye:MOVE0,0:PLOT65,0,ye:MOVE0,0:PLOT6
5,-ex,32:MOVE0,0:PLOT65,ex,32:ye=ye+5
:PLOT65,ex,ye:MOVE0,0:NEXT:ENDPROC
  680 DEFPROCp:*FX12,0
  690 *FX15,0
  700 IFsc=H%:VDU31,8,2,129,141:PRINT
"Congratulations!!";TAB(8,3);:VDU130,
141:PRINT"Congratulations!!":SOUND-1,
279,0,0:SOUND-1,202,0,0:PROCwt(150)
  710 PRINTTAB(8,6);:VDU131,141:PRINT
"Your Score "CHR$129;sc;TAB(8,7);:VDU
131,141:PRINT"Your Score "CHR$129;sc;
TAB(8,10);:VDU134,141:PRINT"High Scor
e "CHR$136;H%;TAB(8,11);:VDU134,141:P
RINT"High Score "CHR$136;H%
  720 RESTORE1440:PROCspk:sc$=STR$sc:
IFLENsc$<>4:REPEATsc$="0"+sc$:UNTILLE
Nsc$=4:IFsc$="0000"SOUND-1,139,0,0:GO
TO760
  730 SOUND-1,VAL(LEFT$(sc$,1))+48,0,
0:SOUND-1,141,0,0:SOUND-1,VAL(MID$(sc
$,2,1))+48,0,0:SOUND-1,140,0,0:SOUND-
1,165,0,0:IFMID$(sc$,3,1)="1"ANDMID$(
sc$,4,1)="0"SOUND-1,264,0,0:GOTO760
  740 IFMID$(sc$,3,1)="1"SOUND-1,VAL(
MID$(sc$,4,1))+32,0,0:SOUND-1,135,0,0
:GOTO760ELSEIFMID$(sc$,3,1)="0"SOUND-
1,VAL(MID$(sc$,4,1))+48,0,0ELSE SOUND
-1,VAL(MID$(sc$,3,1))+32,0,0:SOUND-1,
137,0,0
  750 IFMID$(sc$,4,1)<>"0"SOUND-1,VAL
(MID$(sc$,4,1))+48,0,0:*FX21,0
  760 CLEAR:PROCwt(500):VDU31,2,17,14
1,134:PRINT"Do you want to play again
 ?";TAB(2,18);:VDU141,132:PRINT"Do yo
u want to play again ?":RESTORE1450:P
ROCspk:A=GET:*FX21,8
  770 IFA=78OR A=110PROCrh:*FX21,8
  780 IFA=78OR A=110PRINTTAB(0,22)SPC
8"Good_bye one and all!"SPC11:RESTORE
1460:PROCspk:PROCwt(400):PROCr:END
  790 IFA=89OR A=121:PROCa:GOTO90ELSE
760
  800 ENDPROC
  810 DEFPROCre:*FX21,0
  820 VDU23;10,32,0;0;0;:*FX9,25
  830 *FX10,25
  840 *FX21,8
  850 IFlI<10SOUND-1,229,0,0:SOUND-1,
lI+48,0,0
  860 COLOUR12:PRINTTAB(6,15);"READY!
!":PROCwt(150):PROCi2:PROCs:*FX15,0
  870 ENDPROC
  880 DEFPROCe:*FX21,0
  890 IFy=TRUE PRINTTAB(xo,28);" ":y=
FALSE:ENDPROC
  900 PRINTTAB(xo,yh);" ";TAB(xo,28);
" ":ENDPROC
  910 DEFPROCpa:*FX15,1
  920 *FX12,0
  930 IFINKEY(-81)REPEATUNTILINKEY(-6
5)
  940 IFINKEY(-17):*FX210,1
  950 IFINKEY(-82):*FX210,0
  960 *FX12,100
  970 ENDPROC
  980 DEFPROCff:RESTORE1350:*FX21,0
  990 REPEAT:READV,PP,D:IFV=1V=-15
 1000 SOUND1,V,PP,D:SOUND2,V,PP-48,D:
SOUND3,V,PP+48,D:UNTILD=99:*FX15,1
 1010 ENDPROC
 1020 DEFPROChp:LOCALi:FORi=0TO7:VDU3
1,0,i,154,RND(7)+144:NEXT:PROCcd(0):F
ORi=7TO14:VDU31,0,i,136,153,RND(7)+14
4:NEXT:PROCcd(7):FORi=14TO20:VDU31,0,
i,154,RND(7)+144:NEXT:PROCcd(14):ENDP
ROC
 1030 DEFPROCcd(ee):*FX21,8
 1040 PRINTTAB(4,ee)"x/tj5"SPC12CHR$2
55"/a0"TAB(4,ee+1)CHR$255" /j5  _0"SP
C8CHR$255" j5"TAB(4,ee+2)CHR$255"  j5
p b10p  _p0 "CHR$255" "CHR$34"}__0_p0
__p0"TAB(4,ee+3)CHR$255"  j";
 1050 PRINTCHR$255CHR$163"}j5"CHR$255
CHR$163"}(ss"CHR$255" "CHR$255" _?j7"
CHR$163CHR$255CHR$163CHR$255"j7"CHR$1
63CHR$255TAB(4,ee+4)CHR$255" _z5 "CHR
$255"j5"CHR$255" "CHR$255"j7"CHR$163C
HR$255" "CHR$255" j5j5 "CHR$255" "CHR
$255"j}!?"
 1060 PRINTTAB(4,ee+5)"+!'k5 "CHR$255
":u"CHR$255" ";CHR$255":}!"CHR$255" "
CHR$255"!>!j5 o!?j5"TAB(31,ee+6)"j5":
ENDPROC
 1070 DEFPROCc:VDU29,640;0;:VDU24,-64
0;500;639;1023;:GCOL33,134:CLG:VDU24,
-640;0;639;1023;:GCOL0,6:MOVE-640,500
```



```
:DRAW640,500:FOR6%=-640TO640STEP12:MO
VE6%,500:DRAW8*6%,0:NEXT:VDU26:MOVE10
00,500:6COL0,0:DRAW1030,550:DRAW1060,
500:MOVE1030,550
 1080 DRAW1030,675:MDVE1000,675:MOVE1
060,675:6COL0,3:PLOT85,1000,745:PLDT8
5,1060,745:MOVE800,564:VDU19,10,0;0;:
6COL0,10:VDU5:VDU242,10,8,243:PROCdo(
5):PROCwt(100):PROCdo(0):MOVE165,564:
6COL0,7:VDU224,10,8,227:PROCwt(50)
 1090 PROCdo(5):MDVE300,564:6COL0,0:V
DU228,10,8,241:MOVE370,600:8CDL0,0:VD
U251,252:SOUND16,4,3+RND(3),2:PRDCwt(
50):MOVE370,600:6CDL33,6:VDU251,252:F
ORxs=330TO800STEP30:MOVExs,560:6COL0,
0:VDU255,8:PRDCwt(5):6COL33,6:VDU255:
NEXT:PRINT">"
 1100 PROCwt(10):MOVE840,560:6COL33,6
:PRINT">":MOVE760,600:8COL0,0:VDU253,
254:VDU19,10,9;0;:SOUND1,3,200,20:PRD
Cwt(50):MOVE760,600:8COL33,6:VDU253,2
54:MOVE800,564:6COL0,0:VDU242,10,8,24
3:8COL33,6:MOVE800,564:VDU242,10,8,24
3:6CDL0,1
 1110 SDUND-1,7,0,0:MOVE800,532:VDU24
9,250:PROCwt(200):ENDPROC
 1120 DEFPROCdo(U):8COL0,U:MDVE100,50
0:MOVE250,500:PLOT85,250,700:MOVE100,
700:PLDT85,100,500:IFU=56CDL0,7:PLOT6
9,220,610
 1130 ENDPROC
 1140 DEFPRDCgh:RESTORE1430:PROCspk
 1150 PRINTTAB(1,4);CHR$134"Do you wa
nt to load the High score ?":A=INSTR(
"YNyn",BET$):IFA=2OR A=4ENDPROC
 1160 IFA=1OR A=3VDU28,0,10,39,6:H=OP
ENIN"HIGH":INPUT#H,H%:CLOSE#H:VDU26:E
NDPROC ELSE1150
 1170 DEFPROCrh:RESTORE1470:PRDCspk
 1180 LOCALA:PRINTTAB(1,22);CHR$134"D
o you want to record the High score ?
":A=INSTR("YNyn",6ET$):IFA=2DR A=4END
PROC
 1190 IFA=1OR A=3VDU28,0,24,19,23:H=O
PENOUT"HI8H":PRINT#H,H%:CLOSE#H:CLS:V
DU26:ENDPROC ELSE1180
 1200 DEFPROCas:P%=&A05
 1210 [OPTO:LDA#0:LDX#0:LDA&0A40,X
 1220 .write JSR&FFEE:INX:LDA&0A40,X:
CMP#&FF:BNEwrite
 1230 RTS:]
 1240 P%=&A20
 1250 [OPTO:LDA#0:LDX#0:LDA&0A4A,X
 1260 .write2 JSR&FFEE:INX:LDA&0A4A,X
:CMP#&FF:8NEwrite2
 1270 RTS:]:ENDPROC
 1280 DEFPRDCr:*FX4,0
 1290 *FX12,0
 1300 *FX15,0
 1310 *FX210,0
 1320 ENDPROC
 1330 PRINT'"Please wait...":*TAPE
 1340 FDRI%=0TO TOP-PAGE STEP4:I%!&E0
0=I%!PAGE:NEXT:?&13=?&13-(PA8E-&E00)D
IV256:PA6E=&E00:RUN
 1350 DATA1,53,4,0,0,.25,1,53,4,1,73,
4,0,0,.25,1,73,4,1,81,4,0,0,.25,1,81,
4,1,89,9,1,73,5,0,0,.25,1,53,3.5,0,0,
.25,1,53,3.5,1,73,4,0,0,.25,1,73,4,1,
89,2.5,1,73,2.5,1,89,2.5,1,73,15,0,0,
99
 1360 DATA0,0,0,24,24,0,0,0,28,28,28,
8,60,90,153,153,0,0,0,64,127,63,3,3,0
,0,0,1,127,126,96,96,153,153,24,36,36
,66,36,102,112,120,112,39,124,112,112
,112,1,0,3,7,13,31,13,6,0,128,192,224
,176,248,176,96,7,31,63,111,239,207,2
23,126,224,248,252
 1370 DATA246,247,243,251,254,31,63,6
3,63,31,31,15,14,248,252,252,252,248,
248,240,112,30,60,120,124,62,30,126,2
52,120,60,30,62,124,120,126,63,0,0,0,
0,0,0,0,0,0,0,0,251,170,171,170,0,0,0
,0,190,170,170,170,0,0,0,0,228,164,22
8,160,4,112,112,80
 1380 DATA72,72,72,72,108,7,15,7,34,6
3,7,7,7,7,7,9,9,17,17,17,51,28,60,254
,255,126,63,125,56,255,126,126,60,24,
24,24,126,248,250,253,249,249,250,252
,248,250,253,249,114,124,112,32,248,0
,24,60,24,126,24,60,126,0,0,0,0,0,129
,135,255,0,0,0,0,2
 1390 DATA231,247,255,238,170,170,206
,170,170,234,0,238,170,168,171,170,17
0,174,0,174,170,168,171,170,170,238,0
,165,165,165,229,165,160,165,0
 1400 DATA241,246,252,1
 1410 DATA17,7,31,8,30,32,225,226,32,
255,17,7,31,8,2,237,237,237,237,8,8,8
,8,10,237,229,230,237,8,8,8,8,10,237,
231,232,237,8,8,8,8,10,17,1,237,233,2
34,237,8,8,8,8,10,17,7,237,235,236,23
7,8,8,8,8,10,237,237,237,237,255
 1420 DATA241,246,252,237,267,174,1
 1430 DATA184,275,281,267,257,200,176
,1
 1440 DATA290,257,209,1
 1450 DATA162,289,237,226,1
 1460 DATA265,275,147,254,270,247,1
 1470 DATA184,275,281,267,257,235,143
,176,1
```

THE NUMBER ONE ADVENTURE SERIES

Triffid Software Research

BBC–B RUNEMAGIC SERIES
ADVENTURE ONE
THE SECRET RIVER

BBC-B versions £6-95 each

Triffid Software Research

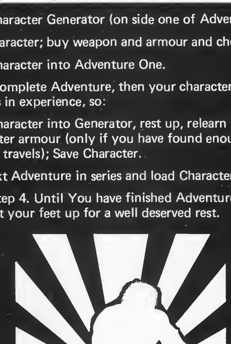
BBC–B RUNEMAGIC SERIES
ADVENTURE TWO
THE WIZARD'S CITADEL

1. Load Character Generator (on side one of Adventure One).
2. Dice Character; buy weapon and armour and choose spells.
3. Load Character into Adventure One.
4. If you complete Adventure, then your character will progress in experience, so:
5. Load Character into Generator, rest up, relearn spells and buy better armour (only if you have found enough money on your travels); Save Character.
6. Buy next Adventure in series and load Character in.
7. Go to step 4. Until You have finished Adventure Eight then put your feet up for a well deserved rest.

Available by M/Order (p+p free)
Send cheques or Postal Orders to:

Triffid Software Research
12 Cullwood Lane
Ashley
New Milton
Hampshire
BH25 5QJ

Available from good computer software stores nationwide



SPECTRUM 48K, ELECTRON and COMMODORE 64 versions coming soon

Send your software to us.
If accepted,
generous royalties offered.

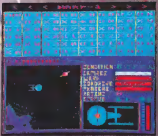
AVAILABLE NOW!
Super colour graphics with six skill levels & 48 screen locations

COMING NOV 1st
BBC/ELECTRON/48K SPECTRUM

arcade fun as Santa delivers the presents on Xmas eve... overcome hostile snowmen, flying snowballs, ice & icicles etc. whilst consuming as much cake & sherry as possible on the way. Hi-res full colour graphics £7.95

Zorakk the Conqueror... a strategical fantasy adventure with full colour graphics. £7.95

Space station Alpha... A full colour space battle with 3D graphics £7.95



Space...the final frontier...
as commander of a federation starship, your mission is to seek out a fellow captain who is lost in space. You must explore strange new galaxies, fight and destroy the enemy and boldly go where no man has gone before.

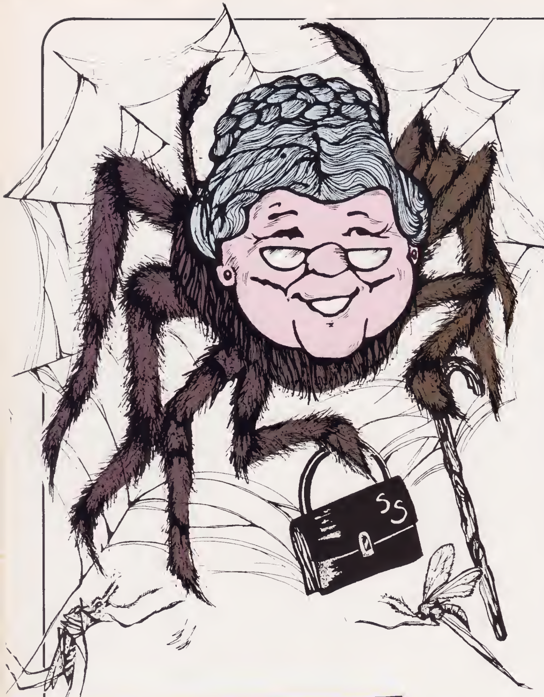
BBC £7.95 ELECTRON £7.95

ICON SOFTWARE

65 High Street, Gosforth, Tyne & Wear. Tel: (091) 2846966

ORDER BY POST Access/Barclaycard or send cheque/P.O.

Available from most major stockists, all good computer shops or direct from us. Dealers contact: Centresoft, Express.



Flee the gardener ...catch the fleas

Help a senior spider round the garden in ALAN SERGEANT's lunchtime lark-about

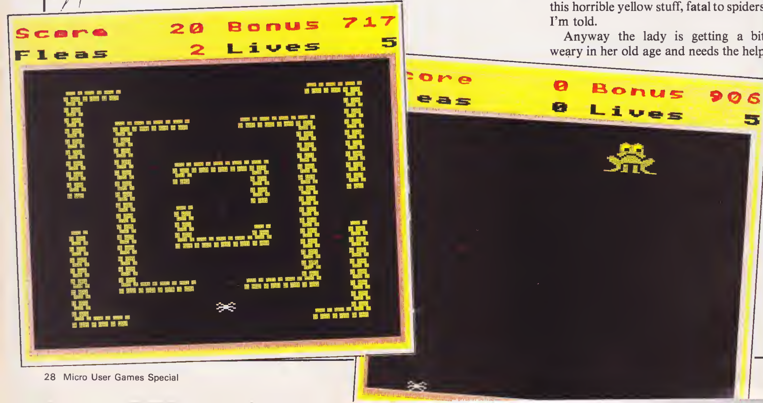
ME mam always told me that there were fairies at the bottom of our garden – well, I checked the other day and believe me it's a load of old rubbish. The place is full of insects and other horrible creepy crawlies.

Down in the bottom corner, just to the left of the garden roller, there's the most gi-normous white lady spider I've ever seen in my life. I can tell she's a lady by the way she keeps smashing everything in sight with her handbag.

Alas and alack, she is now a widow and her daily diet of fresh fleas is getting more difficult to come by, what with inflation as it is, and VAT (vermin added tax) escalating out of all proportion.

To cap it all the gardener, in his wisdom, has sprayed everything with this horrible yellow stuff, fatal to spiders I'm told.

Anyway the lady is getting a bit weary in her old age and needs the help



of your nimble fingers to get her around the garden.

You move Arachnida – she didn't pick the name – round the screen using:

A *(up)* **.** *(left)*
Z *(down)* **,** *(right)*

You must catch a minimum of 20 fleas before you move on to another screen.

There are seven different screens in each round and on the first run through there are a maximum of eight fleas on the screen. This number decreases by one each round to make the game a little more difficult.

You score 10 points for each flea you catch, plus whatever bonus points are left if you clear a screen.

Just be careful of anything yellow – the walls, the frog, the plant pots, and even the gardener's boots are poisonous.

One touch means certain death, although you do have five lives to play with. There is a time limit on each screen so don't hang about 'cos the fleas won't.

PROCEDURES

Most of the procedures are adequately described by their names, but the following rèquire a little more description.

PROCinit_arrays	Initialises all arrays which are called but only once.
PROCstartup	Initialises screen and restores data for music for each round.
PROCinit	Initialises variables and envelopes.
PROCsetup	Initialises certain variables and strings.
PROCsoundoff	Empties the sound queue at "Game over".
PROCtun (P%,D%,DD%)	Creates the music and passes parameters to PROCmusic, P% – offset, D% – note length, DD% – pause length.
PROCmusic	Called throughout the game with updated parameters.
PROCdbl (msg$,X%,Y%,C%)	Double height characters passing the string, X, Y coordinates and colour.
PROCsgl (msg$,Y%,C%)	Single height centralised characters passing the string, Y coordinates and colour.
PROCenv(e%)	Toggles between two envelopes with the same number.
FNp (X%,Y%,C%)	Hit check for colour C% at location X%, Y%.

VARIABLES

BO%	Bonus.
BPITCH%	Base pitch for music.
co%	Count for number of lines scrolled up.
dead%	Number of fleas caught.
F%	Random number for feet to move.
frog%	Count for delay in froghop.
FOOTflag%	Flag for deciding which foot to move on screen 1.
HI%	Hiscore.
lives%	Lives.
MAX%	Maximum number of fleas on screen.
NEWRIGHTX%, NEWLEFTX% **OLDRIGHTX%, OLDLEFTX%** **NEWRIGHTY%, NEWLEFTY%** **OLDRIGHTY%, OLDLEFTY%**	Coordinates of feet.
NEWX%, NEWY% **OLDX%, OLDY%**	Coordinates of spider.
NFX%, NFY% **OFX%, OFY%**	Coordinates of frog.
P%, D%, DD%	Offset from BPITCH%, note length, pause length.
snuffedit%	True or false for dead spider.
SCREEN%	Screen number.
XX%,YY%	Coordinates of teletext spider.

THE following routines are a little more complicated and require further explanation:

PROCtun(P%,D%,DD%) – Writes the music that is passed to PROCmusic. The parameters are passed because without this, each time the PROC is called, the pitch resets.

It is this procedure which keeps the music playing throughout the game by using the ADVAL(–8) test. This checks for free spaces in the channel three sound buffer and, if there are any, continues playing the tune.

PROCscroll — Inserts a space in the bottom right hand corner of the screen forcing it to scroll upwards. If you prefer not to have the scroll, replace PROCscroll in lines 4190, 4260, 4380 and 4490 with CLS and omit lines 4620-4720.

PROCenv(e%) – This is necessary to cater for the 0.1 OS which only allows the user to define four envelopes. This game needs five and gets round this by defining two envelopes with the same number, (1), and toggling between them according to the value of e%.

PROCscoretbl – In order to control the input to the high score table, I have used an OSWORD call set at &FFF1 (line 700), with A% set to 0.

This routine takes a specified number of characters from the currently selected input stream. Input is terminated with a Return, the last character can be deleted and Ctrl+U deletes the entire line.

If the maximum number of characters is exceeded a VDU7 prompts. X% and Y% are the low and high bytes of the address of the parameter block. The parameters are set up using the following:

block%?0 and **block%?1** are the low and high bytes of the start address of the input buffer.

block%?2 is the maximum length of the input stream.

block%?3 and **block%?4** are the minimum and maximum acceptable Ascii values of the input.

***FX12,0** resets the auto repeat delay and ***FX15,1** flushes the keyboard buffer. OSWORD is called and finally the string in **buff%** is allocated to **names$(N)** using the string indirection operator **$**.

From Page 29

```
    5 REM (C) The Micro User
   10 PROCinit_arrays
   20 MODE7
   30 VDU23;8202;0;0;0;
   40 PROCinst
   50 *FX12,2
   60 REM***************************** *
   70 REM   START OF MAIN LOOP
   80 REM*************************
   90 REPEAT
  100 MODE5
  110 PROCstartup
  120 MAX%=8
  130 PROCinit
  140 PROCsetup
  150 REPEAT
  160 PROCscreen1
  170 PROCscreen_num
  180 PROCMUSIC
  190 IF SCREEN%=2 PROCscreen2
  200 IF SCREEN%=3 PROCscreen3
  210 IF SCREEN%=4 PROCscreen4
  220 IF SCREEN%=5 PROCscreen5
  230 IF SCREEN%=6 PROCscreen6
  240 IF SCREEN%=7 PROCscreen7
  250 PROCMUSIC
  260 PROCfleas
  270 REPEAT
  280 PROCtun(P%,D%,DD%)
  290 PROCbonus
  300 FLEA%=FLEA%+1
  310 IF FLEA%=MAX% FLEA%=0
  320 PROCmove_flea
  330 PROCspider
  340 IF FNp(NEWX%,NEWY%,2) snuffedit
%=TRUE:PROCdeadspider:lives%=lives%-1
  350 F%=RND(ABS(12-SCREEN%))
  360 IFF%=1 AND SCREEN%=1 PROCmove_f
eet
  370 IF SCREEN%=7 PROCfroghop
  380 UNTIL dead%>=20 OR snuffedit% O
R BO%<1
  390 IF BO%<1 AND lives%>=1 THEN CLG
:PROCsetup
  400 IF dead%>=20 THEN CLG:SOUND&11,
4,0,40:PROCdelay(2):SC%=SC%+BO%:PROCs
core:PROCsetup:SCREEN%=SCREEN%+1:IF S
CREEN%>7 SCREEN%=1:MAX%=MAX%-1
  410 IF lives%>=1 AND snuffedit% THE
N CLG:PROCscore:PROClives:PROCsetup
  420 UNTIL lives%<1:PROCgameover:MOD
E7:VDU23;8202;0;0;0;:PROCsoundoff:P
ROCscoretbl
  430 CLS
  440 PROChiscores
  450 UNTIL 0
  460 VDU22,7:REPORT:PRINT" at line",
ERL
  470 END
  480 REM****************************
  490 REM      END OF MAIN LOOP
  500 REM****************************
  510 DEFPROCstartup
  520 RESTORE3830
  530 VDU23;8202;0;0;0;
  540 VDU24,20;20;1260;880;
  550 COLOUR130:GCOL0,130:GCOL0,129:C
LS:CLG
  560 VDU24,34;36;1246;866;
  570 GCOL0,128:CLG:GCOL0,1
  580 ENDPROC
  590
  600 REM****************************
***
  610 DEFPROCinit_arrays
  620 DIM block% 6:DIM buff% 20
  630 DIMFLEAS(36),sc%(11),name$(11)
  640 FOR N%=0 TO 11:sc%(N%)=ABS(10-N
%)*100:name$(N%)="The Micro User":NEX
T
  650 ENDPROC
  660
  670 REM****************************
***
  680 DEFPROCinit
  690  X=0:lives%=5:musicoff%=0
  700 OSWORD=&FFF1
  710 BPITCH%=43
  720 FOOTflag%=1
  730 ENVELOPE1,1,20,-20,10,10,0,0,0,
0,0,0,0,0
  740 ENVELOPE2,128,30,-10,0,10,3,0,1
26,-10,0,0,126,0
  750 ENVELOPE3,6,95,0,0,137,0,0,92,0
,0,-84,82,100
  760 ENVELOPE4,1,1,0,0,200,0,0,126,0
,0,-126,126,126
  770 ENVELOPE5,1,43,0,0,100,0,0,126,
0,0,-126,126,126
  780 SC%=0:HI%=0 :SCREEN%=1
  790 PROCchars
  800 PROCfeet
  810 ENDPROC
  820
  830 REM****************************
*
  840 DEFPROCfleas
  850 FOR FLEA%=0 TO MAX%-1:PROCnew_f
lea(FLEA%):NEXT
  860 FLEA%=0
  870 ENDPROC
  880
  890 REM****************************
**
  900 DEFPROCsoundoff
  910 SOUND&11,0,0,0
  920 SOUND&12,0,0,0
  930 SOUND&13,0,0,0
  940 ENDPROC
  950 REM***************************
  960 REM        PLACE FEET
  970 REM***************************
  980 DEFPROCfeet
  990 FOOTflag%=1:VDU5
 1000 NEWRIGHTx%=1800:OLDRIGHTx%=NEWR
IGHTx%
 1010 NEWRIGHTy%=RND(24)*32+64
 1020 OLDRIGHTy%=NEWRIGHTy%
 1030 NEWLEFTy%=NEWRIGHTy%-64
 1040 OLDLEFTy%=NEWLEFTy%
 1050 NEWLEFTx%=NEWRIGHTx%-100:OLDLEF
Tx%=OLDRIGHTx%-100
 1060 MOVE NEWRIGHTx%,NEWRIGHTy%:GCOL
3,2:VDU226,227:MOVE NEWLEFTx%,NEWLEFT
y%:VDU226,227,4
 1070 ENDPROC
 1080
 1090 REM****************************
*
 1100 DEFPROCsetup
 1110 TIME=0
 1120 snuffedit%=FALSE
 1130 dead%=0:FLEA%=0
 1140 frog%=0:W%=40
 1150 SOUND&11,5,17,15
 1160 BO%=0::NEWX%=160:NEWY%=64:VDU5:
MOVENEWX%,NEWY%:GCOL3,3:VDU224
 1170 time_limit=RND(40)+40
 1180 SC$=STR$(SC%):BO$=STR$(BO%):HI$
=STR$(HI%):SCREEN$=STR$(SCREEN%):dead
$=STR$(dead%):lives$=STR$(lives%)
 1190 OLDRIGHTx%=-1500:OLDLEFTx%=-150
0:NEWRIGHTx%=-1500:NEWLEFTx%=-1500
 1200 ENDPROC
 1210
 1220 REM****************************
 1230 DEFPROCbonus
 1240 COLOUR1
 1250 VDU4
 1260 BO%=999-TIME/5
 1270 IF BO%<0 THEN BO%=0
 1280 BO$=STR$(BO%)
 1290 PRINTTAB(19-LEN(BO$),1);" ";BO%
;
 1300 VDU5
 1310 ENDPROC
 1320 REM****************************
**
 1330 REM      CREATE SCREENS 1 TO 7
 1340 REM****************************
**
 1350 DEFPROCscreen1
 1360 VDU4
 1370 COLOUR1:PRINTTAB(0,1)"Score ";T
AB(10-LEN(SC$),1)SC%
 1380 PRINTTAB(11,1)"Bonus ";TAB(20-L
EN(BO$),1)BO%
 1390 COLOUR0:PRINTTAB(0,3)"Fleas ";T
AB(10-LEN(dead$))dead%:PRINTTAB(11,3)
"Lives";TAB(20-LEN(lives$),3)lives%
 1400 VDU5
 1410 ENDPROC
 1420 DEFPROCscreen2
 1430 GCOL0,2
 1440 VDU5
 1450 wall$=STRING$(5,CHR$(228))
 1460 MOVE32,704:PRINTwall$
 1470 MOVE480,704:PRINTwall$
 1480 MOVE920,704:PRINTwall$
 1490 MOVE32,340:PRINTwall$
 1500 MOVE480,340:PRINTwall$
 1510 MOVE928,340:PRINTwall$
 1520 MOVE256,544:PRINTwall$
 1530 MOVE704,544:PRINTwall$
 1540 MOVE256,192:PRINTwall$
 1550 MOVE704,192:PRINTwall$
 1560 ENDPROC
 1570 DEFPROCscreen3
 1580 GCOL0,2
 1590 FORP%=0 TO 10:MOVE RND(34)*32+3
2,RND(26)*32+32:VDU229:NEXT
 1600 FORP%=0 TO 10:MOVE RND(34)*32+3
2,RND(26)*32+32:VDU230:NEXT
 1610 ENDPROC
 1620 DEFPROCscreen4
 1630 GCOL0,2
 1640 MOVE160,700:MOVE350,700:PLOT85,
160,500
 1650 MOVE350,700:MOVE350,500:PLOT85,
160,500
 1660 MOVE930,700:MOVE1114,700:PLOT85
,930,500
 1670 MOVE930,500:MOVE1114,500:PLOT85
,1114,700
 1680 MOVE550,700:MOVE734,700:PLOT85,
550,500
 1690 MOVE550,500:MOVE734,500:PLOT85,
734,700
 1700 MOVE160,400:MOVE350,400:PLOT85,
160,200
 1710 MOVE350,400:MOVE350,200:PLOT85,
160,200
 1720 MOVE930,400:MOVE1114,400:PLOT85
,930,200
 1730 MOVE930,200:MOVE1114,200:PLOT85
,1114,400
 1740 MOVE550,400:MOVE734,400:PLOT85,
550,200
 1750 MOVE550,200:MOVE734,200:PLOT85,
734,400
 1760 ENDPROC
 1770 DEFPROCscreen5
 1780 GCOL0,2
 1790 LOCAL X%,Y%
 1800 FORX%=192 TO 1024 STEP256
 1810 FORY%=100 TO 804 STEP 192
 1820 MOVEX%,Y%:MOVEX%,Y%+96:PLOT85,X
%+92,Y%+96
 1830 MOVEX%,Y%:MOVEX%+92,Y%:PLOT85,X
%+92,Y%+96
 1840 NEXT Y%:NEXTX%
 1850 ENDPROC
 1860 DEFPROCscreen6
 1870 GCOL0,2
 1880 block$=CHR$228
 1890 MOVE160,128:PRINTSTRING$(3,bloc
k$)
 1900 MOVE160,800:PRINTSTRING$(3,bloc
k$)
 1910 FORY%=128 TO 400 STEP32:MOVE160
,Y%:PRINTblock$:NEXT
 1920 FORY%=512 TO 800 STEP32:MOVE160
,Y%:PRINTblock$:NEXT
 1930 MOVE960,128:PRINTSTRING$(3,bloc
k$)
 1940 MOVE960,800:PRINTSTRING$(3,bloc
k$)
 1950 FORY%=128 TO 400 STEP32:MOVE108
8,Y%:PRINTblock$:NEXT
 1960 FORY%=512 TO 800 STEP32:MOVE108
8,Y%:PRINTblock$:NEXT
 1970 FORY%=224 TO 704 STEP32:MOVE320
,Y%:PRINTblock$:NEXT
 1980 FORY%=224 TO 704 STEP32:MOVE928
,Y%:PRINTblock$:NEXT
 1990 MOVE384,224:PRINTSTRING$(3,bloc
k$)
 2000 MOVE384,704:PRINTSTRING$(3,bloc
k$)
 2010 MOVE736,224:PRINTSTRING$(3,bloc
k$)
 2020 MOVE736,704:PRINTSTRING$(3,bloc
k$)
 2030 MOVE512,352:PRINTSTRING$(5,bloc
k$)
 2040 MOVE512,576:PRINTSTRING$(5,bloc
k$)
 2050 FORY%=384TO448STEP32:MOVE512,Y%
:PRINTblock$:NEXT:MOVE768,384:PRINTbl
ock$
 2060 MOVE512,544:PRINTblock$:FORY%=4
80TO576STEP32:MOVE768,Y%:PRINTblock$:
NEXT
 2070 ENDPROC
 2080 DEFPROCscreen7
 2090 GCOL3,2
 2100 NFX%=RND(800)+100:NFY%=RND(640)
+200
 2110 MOVENFX%,NFY%
 2120 VDU231,232,233,10,8,8,8,234,235
,236,10,8,8,8,237,238,239
 2130 MOVENFX%,NFY%:GCOL0,3:VDU9,240
 2140 OFX%=NFX%:OFY%=NFY%
 2150 ENDPROC
 2160 REM****************************
*
 2170 REM         MOVE FROG
 2180 REM****************************
*
 2190 DEFPROCfroghop
 2200 frog%=frog%+1:IF frog% <20ENDPR
OC:ELSE frog%=0
 2210 GCOL3,2
 2220 MOVEOFX%,OFY%
 2230 VDU231,232,233,10,8,8,8,234,235
,236,10,8,8,8,237,238,239
 2240 MOVEOFX%,OFY%:GCOL3,3:VDU9,240
 2250 GCOL3,2
 2260 NFX%=RND(800)+100:NFY%=RND(640)
+200
 2270 MOVENFX%,NFY%
 2280 VDU231,232,233,10,8,8,8,234,235
,236,10,8,8,8,237,238,239
 2290 MOVENFX%,NFY%:GCOL0,3:VDU9,240
 2300 OFX%=NFX%:OFY%=NFY%
 2310 ENDPROC
 2320
 2330 REM****************************
*
 2340 DEFPROCscore
 2350 VDU4
 2360 COLOUR1
 2370 SC$=STR$(SC%)
 2380 PRINTTAB(10-LEN(SC$),1);SC%
 2390 VDU5
 2400 ENDPROC
 2410
 2420 REM****************************
*
 2430 DEFPROClives
 2440 VDU4
 2450 lives$=STR$(lives%)
 2460 PRINTTAB(20-LEN(lives$),3);live
s%
 2470 ENDPROC
 2480
 2490 REM****************************
**
 2500 DEFPROCscreen_num
 2510 VDU4
 2520 SCREEN$=STR$(SCREEN%)
 2530 COLOUR 130:COLOUR0
 2540 PRINTTAB((14-LEN(SCREEN$))DIV 2
,16);"Screen:";SCREEN%:PROCdelay(2):P
RINTTAB((14-LEN(SCREEN$))DIV 2,16);:V
DU4:COLOUR128:PRINTTAB((14-LEN(SCREEN
$))DIV 2,16);SPC(8):COLOUR130:VDU5
 2550 PROCMUSIC
 2560 ENDPROC
 2570
 2580 REM****************************
**
 2590 DEFPROCgameover
 2600 VDU4
 2610 PRINTTAB(6,16)"Game over"
 2620 VDU5
 2630 ENDPROC
 2640
 2650 REM****************************
***
 2660 DEFPROCfleanum
 2670 VDU4
 2680 dead$=STR$(dead%)
 2690 PRINTTAB(10-LEN(dead$),3);dead%
 2700 ENDPROC
 2710 DEFPROCmove_flea
 2720 LOCAL X%,Y%,T%
 2730 X%=FLEAS(FLEA%+0)
 2740 Y%=FLEAS(FLEA%+12)
```

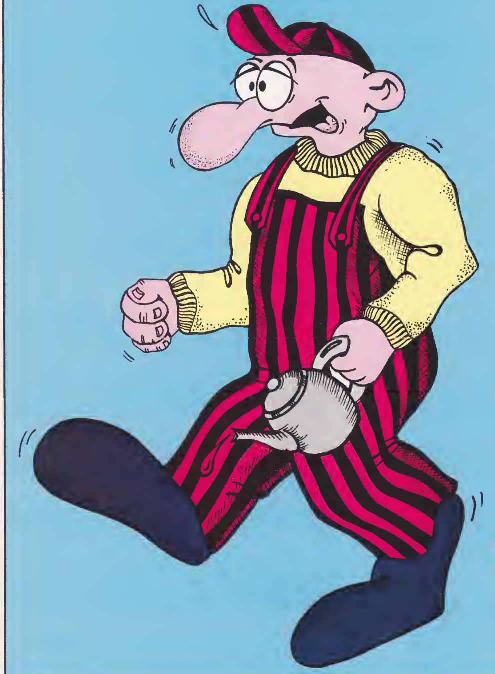
→

From Page 31

```
 2750 T%=FLEAS(FLEA%+24)
 2760 GCDL3,1
 2770 MOVEX%,Y%:VDU5:VDU225
 2780 T%=T%+1
 2790 IF T%>= time_limit THEN SOUND&1
2,2,200,20:PROCnew_flea(FLEA%):ENDPRO
C
 2800 MOVEX%,Y%:VDU5:VDU225
 2810 FLEAS(FLEA%+0)=X%
 2820 FLEAS(FLEA%+12)=Y%
 2830 FLEAS(FLEA%+24)=T%
 2840 ENDPROC
 2850
 2860 REM****************************
 2870 DEFPROCnew_flea(FLEA%)
 2880 8CDL3,1
 2890 REPEAT
 2900 FLEAS(FLEA%+0)=RND(34)*32+32
 2910 FLEAS(FLEA%+12)=RND(26)*32+32
 2920 FLEAS(FLEA%+24)=RND(100)
 2930 UNTIL POINT(FLEAS(FLEA%+0),FLEA
S(FLEA%+12))=0 AND PDINT(FLEAS(FLEA%+
0)+64,FLEAS(FLEA%+12)+32)=0 AND POINT
(FLEAS(FLEA%+0)-8,FLEAS(FLEA%+12)-8)=
0
 2940 MOVEFLEAS(FLEA%+0),FLEAS(FLEA%+
12):VDU5,225
 2950 ENDPROC
 2960
 2970 REM****************************
**
 2980 DEFPROCcrunch
 2990 SOUND&11,1,90,255
 3000 SDUND&10,-15,7,3
 3010 ENDPRDC
 3020 REM****************************
 3030 REM M D V E   F E E T
 3040 REM****************************
 3050 DEFPRDCmove_feet
 3060 FDOTflag%=FDOTflag% EDR 1
 3070 IF FOOTflag%=0 NEWRIGHTx%=NEWRI
GHTx%-200
 3080 IF FOOTflag%=1 NEWLEFTx%=NEWLEF
Tx%-200
 3090 1F NEWRIGHTx%<=-300 NEWRIGHTx%=
0:PRDCfeet:ENDPROC
 3100 IF FDOTflag%=1 GCDL3,2:MOVE NEW
LEFTx%,NEWLEFTy%:VDU226:MDVE NEWLEFTx
%+64,NEWLEFTy%:VDU227:PROCcrunch:MDVE
 OLDLEFTx%,DLDLEFTy%:VDU226:MOVE OLDL
EFTx%+64,OLDLEFTy%:VDU227
 3110 IF FDDTflag%=0 GCDL3,2:MOVE NEW
RIGHTx%,NEWRIGHTy%:VDU226:MOVE NEWRIG
HTx%+64,NEWRIGHTy%:VDU227:PRDCcrunch:
MDVE OLDRIGHTx%,OLDRIGHTy%:VDU226:MOV
E DLDRIGHTx%+64,OLDRIGHTy%:VDU227
 3120 DLDRIGHTx%=NEWRIGHTx%:OLDLEFTx%
=NEWLEFTx%
 3130 ENDPRDC
 3140 REM****************************
*
 3150 REM           MDVE SPIDER
 3160 REM****************************
*
 3170 DEFPRDCspider
 3180 OLDX%=NEWX%:OLDY%=NEWY%
 3190 IF INKEY(-17) musicoff%=-1
 3200 IF INKEY(-102) musicoff%=0
 3210 IF INKEY(-66) NEWY%=OLDY%+32
 3220 IF INKEY(-98)  NEWY%=DLDY%-32
 3230 1F INKEY(-103) NEWX%=DLDX%-32
 3240 1F 1NKEY(-104) THEN NEWX%=DLDX%
+32
 3250 IF NEWX%<34 THEN NEWX%=34
 3260 IF NEWX%>1186 THEN NEWX%=1186
 3270 IF NEWY%<64 THEN NEWY%=64
 3280 IF NEWY%>868 THEN NEWY%=868
 3290 IF NDT(NEWX%=DLDX% AND NEWY%=DL
DY%)MOVE NEWX%,NEWY%:GCOL3,3:VDU224:M
OVE DLDX%,OLDY%:VDU224
 3300 IF POINT(NEWX%+32,NEWY%-16)=2 T
HEN PRDCwhichflea
 3310 ENDPRDC
 3320
 3330 REM****************************
*
 3340 DEF PRDCwhichflea
 3350 LDCALC%,N%
 3360 REPEAT
 3370 1F PD1NT(FLEAS(N%)+32,FLEAS(N%+
12)-16)=2 THEN SOUND0,-15,4,2:GCOL3,1
:VDU5:MDVE FLEAS(N%),FLEAS(N%+12):VDU
225,5:PROCnew_flea(N%):C%=1:SC%=SC%+1
0:dead%=dead%+1:VDU4:PROCscore:PRDCfl
eanum:VDU5
 3380 N%=N%+1
 3390 UNTIL C% OR N%>=MAX%
 3400 ENDPRDC
 3410
 3420 REM****************************
**
 3430 DEFPROCdelay(SECS)
 3440 LOCAL time
 3450 time=TIME
 3460 REPEAT:PRDCMUSIC:UNTILTIME>=tim
e+(100*SECS)
 3470 ENDPROC
 3480
 3490 REM****************************
**
 3500 DEFPRDCchars
 3510 VDU23,224,0,66,165,24,126,153,3
6,66
 3520 VDU23,225,0,0,48,56,24,20,0,0
 3530 VDU23,226,0,63,127,255,255,127,
63,0
 3540 VDU23,227,0,128,206,207,207,206
,128,0
 3550 VDU23,228,123,123,123,0,222,222
,222,222
 3560 VDU23,229,255,255,255,126,126,1
26,60,60
 3570 VDU23,230,60,60,126,126,126,255
,255,255
 3580 VDU23,231,0,0,0,1,1,1,1,1
 3590 VDU23,232,0,66,231,255,24,24,24
,24
 3600 VDU23,233,0,0,0,128,128,128,128
,128
 3610 VDU23,234,1,1,3,3,25,60,62,31
 3620 VDU23,235,255,255,195,189,126,2
55,255,255
 3630 VDU23,236,128,128,192,192,152,6
0,124,248
 3640 VDU23,237,31,15,7,3,3,7,14,124
 3650 VDU23,238,255,128,102,102,102,1
02,102,231
 3660 VDU23,239,248,240,224,192,192,2
24,112,62
 3670 VDU23,240,0,0,0,0,231,231,132,1
32
 3680 VDU23,241,0,0,0,0,99,99,0,0
 3690 VDU23,242,0,0,0,0,0,0,99,99
 3700 VDU23,243,0,0,60,66,129,0,0,0
 3710 ENDPRDC
 3720
 3730 REM****************************
*
 3740 DEFPROCtun(P%,D%,DD%)
 3750 IF musicoff% ENDPROC
 3760 IF ADVAL(-8)<8 ENDPROC
 3770 IF BPITCH%>101 8P1TCH%=43
 3780 READP%,D%,DD%
 3790 SOUND3,-5,BPITCH%+P%,D%
 3800  SOUND3,0,0,DD%
 3810 IF P%=0 AND D%=0 AND DD%=0 REST
DRE 3830:BPITCH%=8PITCH%+5:SOUND3,0,0
,40
 3820 ENDPROC
 3830 DATA 0,8,4,28,4,8,28,8,0,16,4,0
,8,8,0,16,4,0,0,12,0,28,12,0,28,16,8,
48,8,0,28,8,4,28,8,0,36,4,0,28,8,0,20
,4,0,16,12,0,0,8,0,8,16,8
 3840 DATA 36,8,0,32,4,0,36,8,0,44,4,
0,48,8,0,44,4,0,48,8,0,56,4,0,64,8,0,
56,4,0,64,8,0,68,4,0,76,16,8,76,8,0,6
4,4,0,48,8,0,56,4,0,64,8,0,56,4,0,64,
4,0,76,4,16,28,8,0,48,16,0,0,0,0
 3850
 3860 REM****************************
*
 3870  DEFPRDCMUS1C
 3880 PROCtun(P%,D%,DD%)
 3890  ENDPROC
 3900
 3910 REM****************************
*
 3920  DEF FNp(X%,Y%,C%)
 3930 =PDINT(X%,Y%-16)=C% OR POINT(X%
,Y%)=C% DR  POINT(X%+60,Y%)=C% OR POI
NT(X%+32,Y%)=C%
 3940
 3950 REM****************************
**
 3960 DEFPRDCdeadspider
 3970 SOUND&11,3,100,16
 3980 ENDPROC
 3990
 4000 REM******************************
***
 4010 DEFPROCinst
 4020 LOCALXX%,YY%
 4030 FORY%=1 TO 4
 4040 PRINT TAB(6,Y%)CHR$129;CHR$157;
TA8(35,Y%)CHR$156
 4050 NEXT
 4060 PROCdbl("  W H 1 T E  W 1 D O W
",7,2,7)
 4070 RESTORE 4160
 4080 REPEAT
 4090 READXX%,YY%
 4100 PR1NTTA8(XX%,YY%)CHR$255
 4110 UNT1LXX%=24 AND YY%=18
 4120 FORY%=21 TO 24
 4130 PR1NTTAB(6,Y%);CHR$132;CHR$157;
TAB(35,Y%);CHR$156;
 4140 NEXT
 4150 PROCdbl("by Alan Sergeant",11,2
2,7)
 4160 DATA 16,7,25,7,15,8,17,8,20,8,2
1,8,24,8,26,8,14,9,18,9,20,9,21,9,23,
9,27,9,16,10,17,10,19,10,20,10,21,10,
22,10,24,10,25,10,15,11,18,11,19,11,2
0,11,21,11,22,11,23,11,26,11,14,12,17
,12,18,12,19,12,20,12,21,12,22,12,23,
12,24,12,27,12
 4170 DATA 16,13,18,13,19,13,20,13,21
,13,22,13,23,13,25,13,15,14,19,14,20,
14,21,14,22,14,26,14,15,15,18,15,20,1
5,21,15,23,15,26,15,17,16,24,16,17,17
,24,17,17,18,24,18,999,999
 4180 A$=INKEY$(200)
 4190 PROCscroll
 4200 RESTORE 3830
 4210 FORY%=10 TO 13:PR1NTTA8(11,Y%);
CHR$129;CHR$157;TAB(28,Y%);CHR$156;:N
EXT
 4220 PROCdbl(CHR$136+" 1NSTRUCTIONS?
",10,11,3)
 4230 REPEAT :6=6ET
 4240 UNT1L 6=78 OR 6=89
 4250 IF 6=78 THEN ENDPROC
 4260 PROCscroll
 4270 PRINTTA8(14,2)CHR$141;CHR$130;"
ARACHN1DA"
 4280 PR1NTTA8(14,3);CHR$141;CHR$131;
"ARACHN1DA"
 4290 PROCsng("is a lady and lives al
one.",5,3)
 4300 PROCsng("She must survive on a
diet of fleas.",7,3)
 4310 PROCsng("Life won't be easy. Th
e garden walls",9,3)
 4320 PROCsng("and all obstacles, hav
e been painted",11,3)
 4330 PROCsng("with a deadly yellow p
oison.",13,3)
 4340 PROCsng("She must eat 20 fleas
before moving",16,1)
 4350 PROCsng("to a different locatio
n.",18,1)
 4360 PROCsng("Press any key to conti
nue",22,3)
 4370 A=6ET
 4380 PROCscroll
 4390 PROCsng("Move Arachnida around
using these keys",4,3)
 4400 PROCdbl("A     _    UP",11,7,1)
 4410 PROCdbl("Z     _    DOWN",11,9,1
)
 4420 PROCdbl("<     _    LEFT",11,11,
1)
 4430 PROCdbl(">     _    R16HT",11,13
,1)
 4440 PROCdbl("Q/M turn OFF/ON the mu
sic",6,16,1)
 4450 PROCsng("REMEM8ER, touch anythi
ng yellow",19,3)
 4460 PROCsng("or run out of time and
 you lose a life.",21,3)
 4470 PROCsng("Press any key to conti
nue",24,6)
 4480 A=6ET
 4490 PROCscroll
 4500 PROCsng("You score 10 for each
flea collected",3,3)
 4510 PROCsng("plus the bonus left on
 a cleared screen",5,3)
 4520 PROCsng("8 fleas appear on scre
en at the start.",8,3)
 4530 PROCsng("As you succeed on the
first 7 screens",10,3)
 4540 PROCsng("this total is reduced
by one per round.",12,3)

 4550 PROCsng("Don't expect the fleas
 to hang about!",16,3)
 4560 PROCsng("They don't particular1
y like spiders!",19,1)
 4570 PROCsng("Press any key to start
",23,6)
 4580 A=6ET
 4590 CLS
 4600 ENDPROC
 4610
 4620 REM******************************
*
 4630 DEFPROCscroll
 4640 co%=0
 4650 REPEAT
 4660 PR1NTTAB(39,24)" "
 4670 co%=co%+1
 4680 FORDELAY= 1TO 200:NEXT
 4690 UNT1Lco%>16
 4700 VDU31,0,0
 4710 ENDPROC
 4720
 4730 REM******************************
**
 4740 DEFPROCdbl(msg$,X%,Y%,C%)
 4750 FOR N%=0 TO 1:PR1NT TAB(X%,Y%+N
%);CHR$141;CHR$(128+C%);msg$;:NEXT
 4760 ENDPROC
 4770 DEFPROCsng(msg$,Y%,C%)
 4780 X%=(40-LEN(msg$))/2
 4790 PR1NTTAB(X%,Y%)CHR$(128+C%);msg
$;
 4800 ENDPROC
 4810
 4820 REM******************************
***
 4830 DEFPROCscoretbl
 4840 N=0:REPEATN=N+1
 4850 UNTIL SC%>sc%(N) OR N=11
 4860 1F N=11 ENDPROC
 4870 FOR N1=11 TO N STEP-1:sc%(N1)=s
c%(N1-1):name$(N1)=name$(N1-1):NEXT
 4880 sc%(N)=SC%
 4890 CLS
 4900 PROCdbl(CHR$(136)+"Congratulati
ons",10,6,7):PROCsng("Your score is i
n the top ten",12,6):PROCsng("Please
enter your name",14,6)
 4910 PRINTTAB(10,16)CHR$(134)CHR$(15
7)CHR$(129);SPC(18);CHR$(156);TA8(13,
16);
 4920 block%?0=buff% MOD 256
 4930 block%?1=buff% D1V 256
 4940 block%?2=17:REM maximum number
of characters.
 4950 block%?3=32:REM minimum ASC1I v
alue = 32
 4960 block%?4=126:REM maximum ASC11
value = 126
 4970 X%=block% MOD 256
 4980 Y%=block% D1V 256
 4990 A%=0
 5000 PROCdelay(1)
 5010 *FX12,0
 5020 *FX15,1
 5030 CALL OSWORD
 5040 name$(N)=$buff%
 5050 ENDPROC
 5060
 5070 REM******************************
**
 5080 DEFPROChiscores
 5090 REPEAT
 5100 PROCdbl(CHR$(157)+CHR$(131)+"
  White Widow Hall Of Fame.",0,0,1):P
R1NTTAB(38,0);CHR$(156);TA8(38,1);CHR
$(156);
 5110 FOR N%=1 TO 10:PRINT TA8(8,N%*2
+2);CHR$(RND(6)+128);sc%(N%);TA8(16,N
%*2+2);CHR$(RND(6)+128);name$(N%);:NE
XT
 5120 PROCsng("Press space bar for a
new game",24,7)
 5130 *FX15,1
 5140 UNT1L  1NKEY(-99)
 5150 ENDPROC
```

The continuing saga of Bill the Spaceman and his quest for fame

Bill bou bac



By
SIMON
PHIPPS

WE thought he had gone, but Simon Phipps returns yet again with a sequel to the highly exciting Dug-Dig *(Micro User, May 1984)*. This time though, he had incorporated a section of machine code in his latest marvel to make the game a little faster and obviously that much harder.

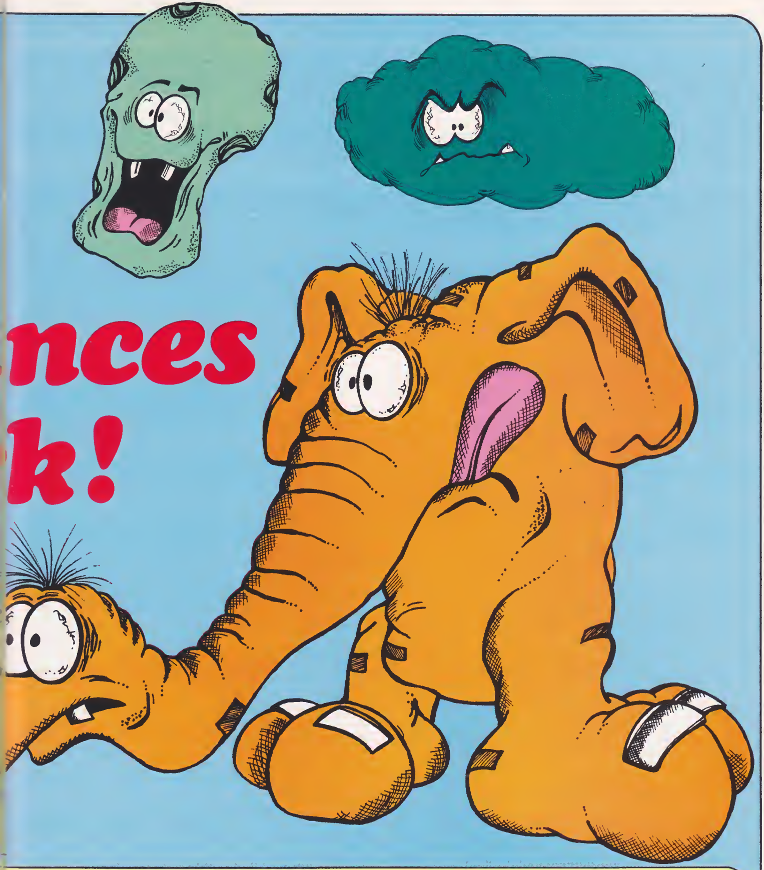
Without further delay we continue the saga of Spaceman Bill and his quest for fame, fortune and an Acorn Electron. (How did that get in?)

There he was with a handful of krystals on a strange asteroid away in the Hilton system. It was quite risky this mining business and Bill was a little worried about the fact that he might not be able to spend all of his profits, what with those nasty Scrugolds and all.

So packing another gem into his starship and zapping a last alien he blasted off for territories unknown and relaxed with a jinantonnixx and his copy of the book that caused him all the trouble in the first place.

He had reached chapter 21,884 by the time the warp drive had cut in when suddenly he found what he thought might be a safer way of making money. Gugvunt Bleuch, author of the best

nces k!



selling "Jewels of the Universe" had made a killing on his book, but Bill was no author, so turning the page he found a section on the sacred Green Teapots of Quarg.

Wasting no time Bill punched the right coordinates into his flight computer and headed for the Beteljoolien star cluster and once more (hopefully) fortune. (He was starting to give up on the fame bit.)

Bill was now certain about everything. Surely nothing dangerous could happen. According to the book the strange teapots were piled high all over the planet Neeguss – the umptifirst moon of the star cluster.

However he was a trifle worried about the fact that the last page of that chapter had been torn out.

As it happened the last page did give a warning along the lines of "Keep away from it mate, it's far too dangerous", but as Bill didn't know this he landed his starship with an ungainly thud on the alien terrain.

He had noticed on the way down that there was a great deal of low cloud cover

From Page 35

and that it moved quickly and erratically across the planet's surface.

Had Bill realised that the clouds were really vaporoids, a strange life form only found in a few video games across the galaxy, he probably wouldn't have landed. But land he did.

Of course that wasn't all that was stopping Bill from having an easy life. The planet's population mainly consisted of Uecchian Cloudhoppers, small cute and cuddly nasties that spent their whole lives jumping up and down, annoying unwary space travellers and consuming the little oxygen that remains in the planet's atmosphere.

Bill dived out of his starship. Taking a large gulp of air he ran towards the teapots. Night was falling and he knew that any collision with a Cloudhopper would mean certain death. Can he make it? Get typing and find out...

It really is quite simply a matter of collecting some objects and returning them to your waiting starship, avoiding numerous space nasties and only having one minute's supply of air to do it in – a theme that has become rather popular recently.

The game hinges around the large section of assembly language (lines 370-520) which controls no less than 18 separate objects on the screen at once.

This technique uses the rather "illegal" method of directly accessing the screen, but seeing that many leading software houses use this method it is quite acceptable.

Just to make the game harder, as each level is progressed night begins to fall (literally) as the stars come out and the Cloudhoppers refuse to jump quite as high into the cold night air...

Have fun...!

VARIABLES

A% Variable used in PROCMAN to find out what must be added to Bill's x coordinate value depending upon the keys pressed and in FNPEEK(x%,y%) to aid in the calculation of the Ascii code of the character position (x%,y%).

C% The value of the address from which the alien and cloud definitions are stored.

CAR% A flag indicating if Bill is carrying a teapot, if zero then no object is being carried.

CODE% Value of the address from which the assembler code is assembled into.

D% Used in the generation of the starfield to calculate the base y coordinate for the stars being plotted.

DIR% Value of the address from which the cloud direction table is stored. A value in the table of 2 indicates the cloud moves left, a value of 1 moves it right.

DY% Value of the address of the start of the alien direction table. A value in this table of 2 moves the alien up, otherwise it will go down.

H% Used in the generation of the starfield.

HI% Table of the hi-bytes of the cloud's positions.

HY% Table of the hi-bytes of the alien's positions.

J% Used to generate the starfield.

KIL% A flag indicating if Bill is still alive, 0 indicates yes, otherwise it's splatto_

LO% Table of the lo-bytes of the cloud's positions.

LY% Table of the lo-bytes of the alien's positions.

P% Program counter, where the assembler code is presently being assembled into.

Q% Bill's x coordinate.

S% Player's present score.

SC% The number of screens successfully completed.

T% General variable used for virtually everything – in fact my favourite variable.

TEA% Number of teapots collected.

X% Start address of table of x values of the cloud's positions.

Y% Start address of table storing Cloudhopper y coordinates.

o% Variable passing on OPT value to the assembler routine.

x% Variable used to pass various x values to PROC$.

y% Variable used as x% to pass various y values to PROC$.

MAN$ Holds the multicoloured definition for the Bill character.

**a$** Used in PROCP(x%,y%,a$) to enable a string to be passed to that procedure.

A Calculates various screen addresses in PROCMEM.

B Execution address of cloud movement routine.

C Execution address of cloud initialisation routine.

D Execution address of alien in initialisation routine.

E Execution address of alien movement routine.

JJ
KK – Loop variables.

X Used in PROCMEM to generate random screen addresses.

Y Used as X in PROCMEM to generate random screen addresses.

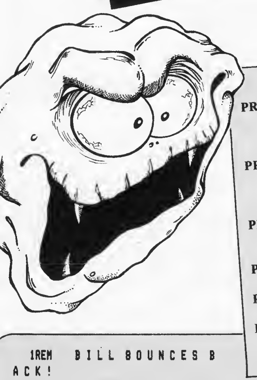
FUNCTIONS

FNPEEK Taken almost exactly from the User Guide, checks the character at the position (x%,y%) and returns its Ascii code.

CONTROLS

Z *Left*
/ *Right*

The game is not very tight on memory and will run quite happily on a standard BBC Micro with 32k. Disc users please note that to run the game with PAGE set to &1900 all unnecessary spaces must be removed and some of the lines may need packing together.



PROCEDURES

PROCASS Contains the assembler code for the aliens and clouds.

PROCBONUS Calculates the bonus points given for completing one screen.

PROCDED Splats Bill about the bottom of the screen.

PROCMAN Moves Bill around the screen.

PROCMEM Initialises the cloud and alien positions.

PROCP Yet again simulates the VDUJ:PRINT TAB(x,y)a$ feature available on the 0.1 OS BBC Micro.

PROCSCORE Prints out the player's score in the box at bottom of the screen.

PROCSCREEN Draws out screen.

PROCTITLE Prints out start up display – "The Micro User presents" etc.

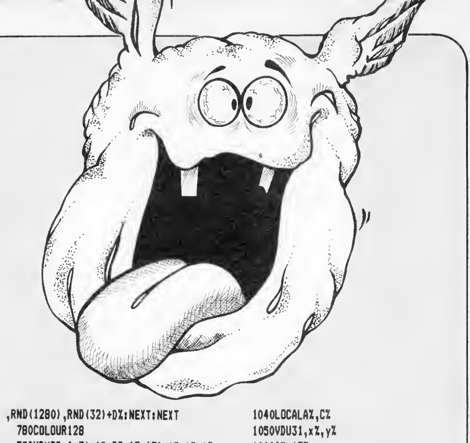
PROCVARI Sets up variables Q%, TEA%, CAR% and KIL%.

PROCcentre Centres text on screen, as passed to procedure by a$.

```
    1REM    B I L L  B O U N C E S  B
 A C K !
    2REM By Simon Phipps (C) The Micr
o User
   10*FX9,7
   20*FX10,7
   30VDU23,224,127,127,127,0,247,247,
247,0
   40VDU23,225,8,28,192,95,125,125,62
,0
   50VDU23,226,8,28,0,0,0,64,32,0
   60VDU23,227,24,24,60,126,255,255,2
55,153,23,228,24,0,0,24,24,0,0,0,23,2
29,0,0,0,0,129,129,153,153
   70VDU23,230,56,56,16,254,16,40,40,
108,23,231,0,0,0,56,16,40,40,0
   80VDU23,232,128,72,1,34,132,52,110
,255
   90ENVELOPE3,1,-10,-37,-89,24,13,46
,43,106,64,64,135,0
  100ENVELOPE1,1,12,-5,15,1,1,60,127,
0,127,127,-127,-127
  110MODE7
  120DIMCODE%&1D0
  130DIMX%8
  140DIMDIR%8
  150DIMHI%8
  160DIMLO%8
  170DIMC%95
  180DIMY%8
  190DIMHY%8
  200DIMLY%8
  210DIMDY%8
  220PROCASS(0):PROCASS(2)
  230MODE2:S%=0:SC%=0:PROCTITLE
  240REPEAT:UNTILGET=32
  241 MEN%=3
  250PROCMEM
  260PROCVARI
  270MODE2:VDU23;8202;0;0;0;
  280VDU28,0,31,19,28,17,128+2,12,20,
26
  290PROCSCREEN
  300?&73=8:TIME=0
  310CALLC:CALLD:REPEAT:CALLB:CALLE:P
ROCMAN:DIR%?(RND(9)-1)=RND(2):UNTILKI
L%<>0OR (TEA%=10AND Q%=0)
  320IFTEA%=10THENPROCBONUS:GOTO250
  330PROCDED
  331 MEN%=MEN%-1:IF MEN%>0 THEN PROC
VARI:PROCMEM:GOTO270
  340COLOUR3:PRINTTAB(5,15)"GAME OVER
":FORKK=0TO4000:NEXT:GOTO230
  350DEFPROCASS(o%)
  360P%=CODE%:[:OPTo%
  370.C:LDX&73:.L1:LDA#C%DIV256:STA&7
1:LDA#C%MOD256:STA&70:LDAHI%,X:STA&81
:LDALO%,X:STA&80:JSRon:DEX:BPLL1:RTS
  380.B:LDX&73:.L2:LDADIR%,X:CMP#2:BE
Q LEFT:LDADIR%,X:CMP#1:BEQ RIGHT:JMP
RET1:.RET:.RET1:DEX:BPLL2:RTS
  390.LEFT:LDAHI%,X:STA&81:LDALO%,X:S
TA&80:JSRoff:LDAX%,X:CMP#0:BEQROFF:CL
C:LDAX%,X:SBC#0:STAX%,X:CLC:LDA LO%,X
:SBC#7:STA LO%,X:LDA HI%,X:SBC#0:STA
HI%,X:LDA#C%DIV256:STA&71:LDA#C%MOD25
6:STA&70:LDAHI%,X:STA&81:LDALO%,X:STA
&80:JSRon:JMPRET
  400.RIGHT:LDAHI%,X:STA&81:LDALO%,X:
STA&80:JSRoff
  410LDAX%,X:CMP#68:BEQLOFF:CLC:LDAX%
,X:ADC#1:STAX%,X:CLC:LDA LO%,X:ADC#8:
STA LO%,X:LDA HI%,X:ADC#0:STA HI%,X:L
DA#(C%+64)DIV256:STA&71:LDA#(C%+64)MO
D256:STA&70:LDAHI%,X:STA&81:LDALO%,X:
STA&80:JSRon:JMPRET:.ROFF:LDA#1:STA D
IR%,X:JMP RET
  420.LOFF:LDA#2:STA DIR%,X:JMP RET
  430.on:LDY#31:.L3:LDA(&70),Y:STA(&8
0),Y:DEY:BPL L3:RTS:.off:LDY#31:.L4:L
DA#0:STA(&80),Y:DEY:BPL L4:RTS
  440.D:LDX&73:.L5:LDA#(C%+32)DIV256:
STA&71:LDA#(C%+32)MOD256:STA&70:LDAHY
%,X:STA&81:LDALY%,X:STA&80:JSRon:DEX:
```

From Page 37

```
BPLL5:RTS
  450.E:LDX&73:.L6
  460LDA DY%,X:CMP#2:BEQ UP:JMP DOWN:
.RET3
  470LDA#(C%+32)DIV256:STA&71:LDA#(C%
+32)MOD256:STA&70:LDAHY%,X:STA&81:LDA
LY%,X:STA&80:JSRon:DEX:BPLL6:RTS
  480.UP:LDAY%,X:CMP#0:BEQ C1:CLC:LDA
LY%,X:SBC#&7F:STA&76:LDAHY%,X:SBC#2:S
TA&77:LDY#31:.C2:LDA(&76),Y:CMP#0:BNE
 C1:DEY:BPLC2:CLC:LDAY%,X:SBC#0:STAY%
,X:LDAHY%,X:STA&81:LDALY%,X:STA&80:JS
Roff:CLC:LDA&76:STALY%,X:LDA&77:STAHY
%,X:JMP RET3:.C1
  490LDA#1:STA DY%,X:JMP RET3
  500.DOWN:LDAY%,X:CMP#27:BEQ D10:CLC
:LDALY%,X:ADC#&80:STA&76:LDAHY%,X:ADC
#2:STA&77:LDY#31:.D2:LDA(&76),Y:CMP#0
:BNE D1:DEY:BPLD2:CLC:LDAY%,X:ADC#1:S
TAY%,X:LDAHY%,X:STA&81:LDALY%,X:STA&8
0:JSRoff:CLC:LDA&76:STALY%,X:LDA&77:S
TAHY%,X:JMPRET3
  510.D1:LDA#2:STA DY%,X:JMP RET3
  520.D10:JMP D1
  530RTS:]:ENDPROC
  540DEFPROCMEM
  550FORT%=0TO8
  560X=RND(60)
  570Y=T%+15
  580A=&3000+Y*640+X*8
  590X%?T%=X
  600HI%?T%=ADIV256
  610LO%?T%=AMOD256
  620DIR%?T%=RND(2)
  630Y=25:X=T%*2+1:X=X*4
  640Y%?T%=Y
  650A=&3000+Y*640+X*8
  660HY%?T%=ADIV256
  670LY%?T%=AMOD256
  680DY%?T%=1
  690MAN$=CHR$18+CHR$0+CHR$7+CHR$230+
CHR$8+CHR$18+CHR$0+CHR$4+CHR$231+CHR$
4
  700NEXT
  710RESTORE:FORT%=0TO95:READC%?T%:NE
XT:ENDPROC
  720ENDPROC
  730DATA0,0,21,63,63,63,21,0,0,63,63
,63,62,63,63,62,0,0,63,63,63,61,62,40
,0,0,0,42,62,63,42,0
  740DATA85,149,1,1,1,1,0,0,1,19,63,2
13,35,0,19,1,34,51,63,213,19,0,51,2,1
70,106,34,34,34,34,0,0
  750DATA0,0,0,21,61,63,21,0,0,21,63,
63,63,62,61,20,0,42,63,63,61,63,63,61
,0,0,42,63,63,63,42,0
  760DEFPROCSCREEN
  770VDU19,14,0;0;:COLOUR142:VDU28,0,
SC%,19,0,12,17,128:FORH%=0TOSC%:D%=(3
1-H%)*32-4:GCOL0,7:FORJ%=0TO10:PLOT69
```



```
,RND(1280),RND(32)+D%:NEXT:NEXT
  780COLOUR128
  790VDU28,0,31,19,28,17,130,12,17,12
8,26
  800COLOUR4:COLOUR134
  810PRINTTAB(0,28)STRING$(60,CHR$224
);
  820COLOUR128
  830COLOUR7:PRINTTAB(7,31);"000000";
  840FORT%=0TO9:MOVE1216,(31-T%-15)*3
2-4:VDU5,18,0,2,225,8,18,0,3,226,4:NE
XT
  850MOVE0,256:VDU5,18,0,3,227,18,0,2
,8,228,18,0,1,8,229,4
  860PROCP(0,27,MAN$)
  861FORT%=1 TO MEN%:PROCP(16+T%,31,M
AN$):NEXT
  870ENDPROC
  880DEFPROCP(x%,y%,a$):VDU5:MOVEx%*6
4,(32-y%)*32-4:PRINT;a$;:VDU4:ENDPROC
  890DEFPROCMAN
  900*FX15,1
  910COLOUR7:PRINTTAB(0,31);" ";(60-T
IMEDIV100);" ";
  920A%=INKEY(-98)-INKEY(-105)
  930PRINTTAB(Q%,27);" ";
  940IFFNPEEK(Q%,26)<>32THENKIL%=1
  950IFA%=-1ANDQ%<>0THENQ%=Q%-1:IFFNP
EEK(Q%,27)<>32THENKIL%=1
  960IFA%=1ANDQ%<>19THENQ%=Q%+1:IFFNP
EEK(Q%,27)<>32THENKIL%=1
  970IFA%<>0SOUND&11,1,65,1
  980PROCP(Q%,27,MAN$)
  990IFQ%=19ANDCAR%=0THENCAR%=1:SOUND
1,-15,160,1:TEA%=TEA%+1:PRINTTAB(19,1
5+TEA%);" ";
 1000IFQ%=0ANDCAR%=1THENCAR%=0:SOUND1
,-15,35,1:S%=S%+10:PROCSCORE
 1010IFTIME>6000THENKIL%=1
 1020ENDPROC
 1030DEFFNPEEK(x%,y%)
 1040LOCALA%,C%
 1050VDU31,x%,y%
 1060A%=135
 1070C%=USR(&FFF4)
 1080C%=C%AND&FFFF
 1090C%=C%DIV&100
 1100=C%
 1110DEFPROCVARI:Q%=0:TEA%=0:CAR%=0:K
IL%=0:ENDPROC
 1120DEFPROCSCORE:COLOUR7:PRINTTAB(7,
31);STRING$(6-LEN(STR$(S%)),"0");S%;:
ENDPROC
 1130DEFPROCDED:SOUND3,3,119,45:FORKK
=0TO100:CALLB:CALLE:COLOUR1:PRINTTAB(
Q%,27);CHR$232:FORJJ=0TO20:NEXT:NEXT:
ENDPROC
 1140DEFPROCBONUS:T%=60-TIMEDIV100:CO
LOUR3:PRINTTAB(3,15)"TIME BONUS=";T%*
10*(SC%+1):SC%=SC%+1:IFSC%=15SC%=14
 1150S%=S%+T%*SC%:PROCSCORE:FORKK=0TO
4000:NEXT:ENDPROC
 1160DEFPROCTITLE
 1170VDU23;8202;0;0;0;
 1180PRINT''
 1190COLOUR5
 1200PROCcentre("The Micro User")
 1210PRINT''
 1220COLOUR1
 1230PROCcentre("presents")
 1240PRINT''''
 1250COLOUR11
 1260PROCcentre("BILL BOUNCES BACK!")
 1270PRINT'':COLOUR1
 1280PROCcentre("by")
 1290PRINT'':COLOUR10
 1300PROCcentre("Simon Phipps")
 1310COLOUR15:PRINTTAB(0,30);:PROCcen
tre("PRESS SPACE TO PLAY")
 1320ENDPROC
 1330DEFPROCcentre(a$):PRINTTAB(10-LE
N(a$)/2);a$:ENDPROC
```

Three programs everyone's talking about!

MICRO OLYMPICS

Play the most exciting, challenging Olympic simulation of them all!

MICRO OLYMPICS is riding high in the charts – and no wonder. You are challenged to beat the world record in 11 realistic track and field events, with the packed stadium cheering you on to victory – or defeat! Just like the real thing, you'll want to keep on improving your sporting skill until you're up there among the greats!

Turn your home micro into a true office workhorse!

MINI OFFICE is the most incredibly priced introduction to business computing ever offered. The package contains four full-scale programs – word processor, database manager, spreadsheet and graphics. They include advanced features not yet available on programs costing many times as much. Plus a concise 32-page how-to-do-it booklet.

The Magic Sword
Kristin and Martin Hollis
The computer Adventure of Princess Poppy and the Frog Prince
A Database publication
The Magic Sword

Let your children embark on a magical micro adventure!

THE MAGIC SWORD breaks new ground in programming for the young – a complete adventure on cassette accompanied by a 48-page full colour storybook. Stunning sound and colourful graphics bring a completely new dimension to the classic computer adventure. And there's a very helpful animated compass to help you find your way around!

Send for them today

Please supply software as ticked in price boxes

□ I enclose cheque made payable to
Database Publications Ltd. for £ ____________

I wish to pay by □ Access □ Visa

No. ______________________ Expiry date ________

Name ______________________

Address ______________________

____

Signed ______________________

Post to: Database Publications, Europa House,
68 Chester Road, Hazel Grove, Stockport SK7 5NY.

	Micro Olympics	Mini Office	Magic Sword
BBC 'B' cassette	£5.95 □	£5.95 □	£8.95* □
Electron cassette	£5.95 □	£5.95 □	
BBC 40 track disc	£7.95 □	£7.95 □	£9.95 □
BBC 80 track disc	£7.95 □	£7.95 □	£9.95 □
Spectrum cassette	£5.95 □	Coming soon	£8.95* □
Commodore cassette	£5.95 □	Coming soon	

* Versions for both machine on same cassette

Electronequip

(Authorised BBC Micro Dealer, and Econet service centre)

	BBC Cessette Software	Exc VAT	Inc VAT
BCAAFRAK	Aerdverk Frek (Ceveman game)	7.00	8.05
BCAAZALA	Aardverk Zelage	7.00	8.05
BCAOFOOT	Addictive Footbell Manager	5.87	6.75
BCAFBOUN	A & F Software Bouncer	5.92	6.80
BCAFCHIC	A & F Softwere Chuckie Egg	5.84	6.72
BCAFCYLO	A & F Softwere Cylon Atteck	5.84	6.72
BCAFFRDG	A & F Softwere Frogger	5.92	6.60
BCAFPAIN	A & F Softwere Peinter	5.92	6.60
BCAFSHRI	A & F Softwere Shrinking Prolessor	5.92	6.80
BCAIAOVE	Adventure Internetionel Adventurelend	7.35	6.46
BCAIPIRA	Adventure International Pirete Adventure	7.35	8.46
BCAISECR	Adventure Internationel Secret Mission	7.35	8.46
BCAIVOOO	Adventure Internationel Voodoo Castle	7.35	8.46
BCAKCHIL	ASK Children from Spece (XBE16)	7.35	8.46
BCAKCRAN	ASK Cranky (XBE17)	7.35	8.46
BCAKFACE	ASK Fecemeker (XBE10)	7.35	8.46
BCAKHIDE	ASK Hide & Seek (XBE11)	7.35	6.46
BCAKLETS	ASK Let's Count (XBE12)	7.35	6.46
BCAKNUMC	ASK Number Cheser (XBE15)	7.35	6.46
BCAKNUMG	ASK Number Gulper (XBR13)	5.62	6.70
BCAKNUMP	ASK Number Puzzler (XBE14)	7.35	6.46
BCAKWDRD	ASK Words, Woids, Words (XBE19)	7.35	8.46
BCALBUGB	Alligeta Software Bug Blaster	5.68	6.76
BCALCOSM	Alligeta Software Cosmic Asteroids	4.39	5.05
BCALDAMB	Alligete Softwere Dambusters	5.87	6.75
BCALEAGL	Alligete Softwere Eegle Empire	5.67	6.75
BCALFRUI	Alligete Softwere Fruit Mechine (B006)	4.39	5.05
BCALUNCL	Alligete Softwere Uncle Claude	7.00	8.05
BCAMALGE	Ampelsoft Cheshire Cat Algebre 0 level	5.13	5.90
BCAMARIT	Ampelsoft Cheshire Cet Arithmetic 0 level	5.13	5.90
BCAMCALC	Ampelsoft Cheshire Cet Celculus 0 level	5.13	5.90
BCAMCONS	Ampelsoft Cheshire Cet Construction "0"	5.13	5.90
BCAMGRAP	Ampelsoft Cheshire Cet Graphs 0 level	5.13	5.90
BCAMTRIG	Ampalsoft Cheshire Cet Trigonometry "0"	5.13	5.90
BCAS100P	Acornsoft 100 Programs for BBC Micro	8.50	9.78
BCASAACT	Acornsoft Arcade Action (SBGD6)	6.80	10.12
BCASARCA	Acornsoft Arcediens (SBG14)	7.35	8.46
BCASAVIA	Acornsoft Avietor flight simuletor SBG02	11.05	12.71
BCASBUSI	Acornsoft Business Gemes (SBE03)	7.35	8.46
BCASCARO	Acornsoft Carousel (SBG24)	7.35	6.46
BCASCAST	Acornsoft Castle of Riddles (SBG17)	7.35	6.46
BCASCHAN	Acornsoft Chemical Analysis (SBE12)	10.20	11.73
BCASCHES	Acornsoft Chess (SBG10)	7.35	8.46
BCASCHSI	Acornsoft Chemicel Simuletions (SBE13)	10.20	11.73
BCASCHST	Acornsoft Chemicel Structures (SBE14)	10.20	11.73
BCASCOLL	Acornsoft/ICL Collectors Cat SBX05	7.35	8.46
BCASCOUN	Acornsoft Countdown to Doom (SBG19)	7.35	8.46
BCASCRAZ	Acornsoft Crezy Trecer (SBG26) (B)	7.35	6.46
BCASCREA	Acornsoft Creative Graphics (SBX01)	7.35	6.46
BCASOESK	Acornsoft Desk Oiary (SBB01)	7.35	8.46
BCASORAU	Acornsoft Oraughts & Reversi (B) (SBG20)	7.35	8.46
BCASOROG	Acornsoft Orogne (SBG27)	7.76	8.95
BCASELIT	Acornsoft Elite (SBG3B)	12.35	14.20
BCASEXAM	Acornsoft Exeminer Cessette (SBE17) (B)	7.35	6.46
BCASFORE	Acornsoft Forecest (SBB02)	6.60	10.12
BCASFORT	Acornsoft Forth Cessette (SBL01)	12.45	14.32
BCASFREE	Acornsoft Free Fell (SBG2B)	7.35	6.46
BCASFREN	Acornsoft French (Linkword) (SBX13)	11.05	12.71
BCASGERM	Acornsoft Germen (Linkword) (SBX1B)	11.05	12.71
BCASGRAO	Acornsoft Grephics on the BBC	8.50	9.76
BCASGRAP	Acornsoft Grephs & Cherts (SBX02)	7.35	6.46
BCASHOOK	A'soft/ICL Hooked on Numbers (SBX0B)	7.35	6.46
BCASHOPP	Acornsoft Hopper (SBG23)	7.35	8.46
BCASITAL	Acornsoft Itelien (Linkword) (SBX14)	11.05	12.71
BCASJARS	Acornsoft Jers (SBE15)	6.83	10.16
BCASJCBD	Acornsoft JCB Oigger (SBG09)	7.35	8.46
BCASKING	Acornsoft Kingdom of Hemil (SBG25)	7.35	8.45
BCASLISO	Acornsoft Lisp Oemo Progs SBL09	7.35	8.46
BCASLISP	Acornsoft Lisp Cassette (SBL02)	12.45	14.32
BCASMAGI	Acornsoft Magic Garden (B) (SBX04)	7.35	8.46
BCASMEMB	Acornsoft/ICL M'ship Meneger (SBX0B)	7.35	8.46
BCASMETE	Acornsoft Meteors (SBG13)	7.35	8.46
BCASMETM	Acornsoft Meteor Mission (SBG29)	7.78	8.95
BCASMICR	Acornsoft Microtext (SBL04)	36.65	42.37
BCASMISB	Acornsoft Missile Base (SBG1B)	7.35	8.46
BCASMISS	Acornsoft Missing Signs (SBE09)	8.83	10.16
BCASMONS	Acornsoft Monsters (SBG03)	7.35	8.4B
BCASNUMB	Acornsoft Number Belance (SBE0B)	6.83	10.16
BCASONET	Acornsoft/ICL One to Nine (SBX07)	7.35	8.46
BCASPEEK	Acornsoft Peeko-Computer Peck (SBE02)	7.35	6.46
BCASPERS	Acornsoft Personel Money M'ment SBB5	8.60	10.12
BCASPHIL	Acornsoft Philosophers Ouest (SBGD1)	7.35	8.46
BCASPICT	Acornsoft Picture Meker (SBXD3)	7.35	8.46
BCASPLAN	Acornsoft Plenetoid (Oelender) (SBG15)	7.35	6.46
BCASROCK	Acornsoft Rocket Reid (SBG05)	7.35	8.46
BCASSENT	Acornsoft Sentence Sequencing (SBE07)	6.B3	10.16
BCASSINV	Acornsoft Super Inveders (SBG16)	7.35	8.46
BCASSLID	Acornsoft Sliding block puzzles (SBG12)	7.35	8.46
BCASSNAP	Acornsoft Snapper (SBG04)	7.35	6.46
BCASSNOO	Acornsoft Snooker (SBG21)	7.35	6.46
BCASSPAN	Acornsoft Spanish (Linkword) (SBX15)	11.05	12.71
BCASSPAS	Acornsoft S Pascel (SBLDB)	12.45	14.32
BCASSPHI	Acornsoft Sphinx Adventure (SBG07)	7.35	8.46
BCASSTAR	Acornsoft Starship Commend (SBG22)	7.35	8.46
BCASTETR	Acornsoft Tetrepod (SBG32)	7.76	8.95
BCASTREE	Acornsoft Tree of Knowledge (SBE04)	7.35	8.46
BCASTURT	Acornsoft Turtle Grephics (SBL07)	12.45	14.32
BCASVIEW	Acornsoft View Printer Driver (SBB04)	7.35	8.45
BCASVOLC	Acornsoft Volceno (SBG33)	7.76	8.95
BCASWDRH	Acornsoft Word Hunt (SBE05)	6.83	10.16

	BBC Cassette Software	Exc VAT	Inc VAT
BCASWORS	Acornsoft Word Sequencing (SBED6)	8.83	10.16
BCBEAVM	BES Animel Vegetable Minerel (XBE03)	6.63	7.62
BCBEHAPL	BES Heppy Letters (XBED7)	6.63	7.62
BCBEHAPN	BES Heppy Numbers (XBED4)	6.63	7.62
BCBEMAPR	BES Mep Relly (XBE0B)	6.63	7.62
BCBETIM1	BES Timemän One (XBE05)	6.63	7.62
BCBETIM2	BES Timemen Two (XBE06)	6.63	7.62
BCBEWDRH	BES Wordheng (XBE02)	6.63	7.62
BCBEWDRW	BES World-Wise (XBE01)	6.63	7.62
BCBSBEYD	BBCsoft Beyond Besic Cessette (B)	8.50	9.76
BCBSCANY	BBCsoft Canyon (B)	7.39	8.50
BCBSORAW	BBCsoft Drewing	7.40	8.50
BCBSORWH	BBCsoft Dr Who The First Adventure (B)	7.39	8.50
BCBSEARL	BBCsoft Early Leerning	7.40	8.50
BCBSFUNG	BBCsoft Fun Gemes	7.40	8.50
BCBSGAMS	BBCsoft Gemes of Stretegy	7.40	B.50
BCBSHOMF	BBCsoft Home Finence	7.40	8.50
BCBSMAKM	Meking the most of your Micro (Inc book)	9.57	11.01
BCBSMOTO	BBCsoft Motorists' Log	7.40	8.50
BCBSPAIN	BBCsoft Painting	7.40	8.50
BCBSPRO1	BBCsoft Progrems Vol 1	7.40	8.50
BCBSPRD2	BBCsoft Progrems Vol 2	7.40	8.50
BCBSRECO	BBCsoft Record Keeper (B)	10.20	11.73
BCBSTAXC	BBCsoft Texcelc (B)	6.16	9.38
BCBSTDDL	BBCsoft Tool Box (B)	15.52	17.85
BCBSVUTY	BBCsoft VU Type (Typing Tutor) (B)	11.90	13.69
BCBSWHIT	BBCsoft White Knight Merk Eleven (B)	6.50	9.78
BCBUTWIN	Bug-Byte Twin Kingdom Valley	7.02	8.07
BCCCANDR	Computer Concepts Android Atteck	6.61	7.60
BCCCASTE	Computer Concepts Asteroid belt	5.87	6.75
BCCCCHES	Computer Concepts Chess	6.61	7.60
BCCCLDGO	Computer Concepts Logo II	6.50	9.76
BCCCSNAK	Computer Concepts Sneke	5.67	6.75
BCCCSPAC	Computer Concepts Specehewks	5.87	6.75
BCCCSWAR	Computer Concepts Swerm	6.61	7.60
BCOS747	Ooctor Soft 747	5.87	6.75
BCGEBEEB	Gemini Beebcelc (Spreadsheet)	14.75	16.96
BCGEBEEP	Gemini Beebplot (Greph Plot)	14.75	16.96
BCGECATE	Gemini Caterpiller	7.35	8.46
BCGGFAIR	Griffin & George Feirshere	7.35	8.46
BCGGGETS	Griffin & George Getset	7.35	8.46
BCGGNUMB	Griffin & George Numberfun	7.35	8.46
BCGGTABL	Griffin & George Teblesums	7.35	8.46
BCGGWORG	Griffin & George Wordgames	7.35	8.46
BCGGWORS	Griffin & George Wordspell	7.35	8.46
BCGHTALP	Good Houseke'g Mr T's Alphebet	9.57	11.01
BCGHTMEA	Good Houseke'g Mr T's Meesuring Games	9.57	11.01
BCGHTMON	Good Houseke'g Mr T's Money Box	9.57	11.01
BCGHTNUM	Good Houseke'g Mr T's Number Games	9.57	11.01
BCGHTSHA	Good Houseke'g Mr T's Shepe Geme	9.57	11.01
BCGHTTEL	Good Houseke'g Mr T Tells the Time	9.57	11.01
BCHOAOVB	Honeyfold Advanced Besic course	12.10	12.71
BCHOBEGA	Honeyfold Beginners Assembly Language	12.10	12.71
BCHOBEGB	Honeyfold Beginners Besic course	12.10	12.71
BCIBCRIM	Iven Berg Crime & Oetection Ouiz (XBX02)	9.35	10.75
BCIBDATI	Ivan Berg Oeting Geme (XBX0B)	9.35	10.75
BCIBHIST	Iven Berg History Ouiz (XBX04)	9.35	10.75
BCIBIDO	Iven Berg 'I 0o' (XBX07)	9.35	10.75
BCIBMUSI	Iven Berg Music Ouiz (XBX03)	9.35	10.75
BCIBROYA	Ivan Berg Royelty Ouiz (XBX06)	9.35	10.75
BCIBSCIE	Iven Berg Science Fiction Ouiz (XBX05)	9.35	10.75
BCIBTHER	Ivan Berg Theatre Ouiz (XBX01)	9.35	10.75
BCIJHYPE	IJK Hyperdrive for BBC (13)	4.60	5.52
BCIJINVA	IJK Inveders for BBC (9)	5.54	6.37
BCIJSTRA	IJK Stratobomber for BBC (14)	5.54	6.37
BCIJSUPE	IJK Super Hangmen	3.32	3.82
BCJDYSTI	Joystick Utility program lor BBC	5.06	5.82
BCKAGALA	Kansas Galactic Firebird	7.01	8.06
BCKOFREA	Kosmos Softwere French Mistress A	7.35	8.46
BCKOFREB	Kosmos Softwere French Mistress B	7.35	8.46
BCKOGERA	Kosmos Softwere Germen Master A	7.35	8.46
BCKOGERB	Kosmos Software Germen Mester B	7.35	8.46
BCKOSPAA	Kosmos Software Spanish Tutor A	7.35	8.46
BCKOSPAB	Kosmos Soltware Spanish Tutor B	7.35	6.46
BCL9COLO	Level 9 Colossel Adventure (B)	7.32	8.42
BCL9OUNG	Level 9 Oungeons (B)	7.32	6.42
BCL9LORO	Level 9 Lords ol Time (b)	7.32	8.42
BCL9SNOW	Level 9 Snowbell (B)	7.32	8.42
BCMHHOBB	Melbourne House Hobbit	11.05	12.71
BCMSOUIC	Mirrorsoft Ouick Thinking (B)	5.13	5.90
BCOCMRWI	Ocean Mr. Wimpy	5.10	5.87
BCPAFORT	Pace Softwere Fortress	6.61	7.60
BCPPADVE	Progrem Power Adventure (B)	5.67	6.75
BCPPALID	Program Power Alien Destroyers (Inveder)	5.91	8.79
BCPPALIE	Program Power Alien Swirl	5.13	5.90
BCPPASTE	Program Power Asteroid Storm (B)	5.67	6.75
BCPPASTR	Progrem Power Astro Nevigetor	4.21	4.64
BCPPBANO	Program Power Bendits et 3 o'clock	5.14	5.91
BCPPBARR	Program Power Berrege for BBC	5.13	5.90
BCPPBEEB	Program Power Beeb-Tote (B)	4.40	5.06
BCPPBUMB	Program Power Bumble Bee (B)	5.67	6.75
BCPPCAVA	Progrem Power Caveman Adventure	5.06	5.62
BCPPCHEM	Program Power Chemistry	5.13	5.90
BCPPCHES	Progrem Power Chess	5.67	6.75
BCPPCOSM	Progrem Power Cosmic Combet (B)	5.14	5.91
BCPPCOWB	Program Power Cowboy Shoot-out	5.06	5.62
BCPPCROA	Progrem Power Croaker	5.87	6.75
BCPPCYBE	Program Power Cybertron Mission (B)	5.87	6.75
BCPPOANG	Program Power Oenger! UXB	5 B7	6.75
BCPPOEMO	Progrem Power Oemon Decoretor (Painter)	5.06	5.62

BUY 3 CASSETTES GET 1 FREE

	BBC Cassette Software	Exc VAT	Inc VAT
BCPPOIS	Progrem Power BBC Oisessembler	5.06	5.82
BCPPOUNE	Program Power Oune Rider	6.22	7.15
BCPPELDO	Progrem Power Eldorado Gold (Adventure)	5.06	5.62
BCPPESCA	Progrem Power Escepe from M'bese Alpha	5.87	6.75
BCPPFFAC	Program Power Felix in the Fectory	5.87	6.75
BCPPFFRU	Program Power Felix & the Fruit Monsters	5.87	6.75
BCPPGHOU	Progrem Power Ghouls	5.87	6.75
BCPPHELL	Progrem Power Hell Oriver (B)	5.B7	6.75
BCPPINTE	Program Power Intergelectic Treder (B)	6.61	7.60
BCPPJETP	Progrem Power Jet Power Jeck	5.87	6.75
BCPPKILL	Progrem Power Killer Gorilla	5.87	6.75
BCPPLABC	Progrem Power Lebyrinths of le Coshe	5.87	6.75
BCPPLASE	Progrem Power Leser Commend	5.87	6.75
BCPPMART	Progrem Power Mertien Atteck (B)	5.B7	6.75
BCPPMINE	Progrem Power Mine (B)	5.87	6.75
BCPPMOON	Progrem Power Moon Reider	5.87	6.75
BCPPNEME	Progrem Power Nemesis (B)	5.B7	6.75
BCPPPHYS	Progrem Power Physics	5.06	5.82
BCPPPOKE	Progrem Power Poker Oice (B)	4.40	5.06
BCPPPOSI	Progrem Power Positron (B)	5.14	5.91
BCPPREV2	Progrem Power Reversi (32K)	4.21	4.64
BCPPROUL	Progrem Power Roulette for BBC	4.21	4.84
BCPPSEEK	Progrem Power Seek	5.06	5.82
BCPPSPAC	Program Power Space Jailer (B)	5.14	5.91
BCPPSTOC	Progrem Power Stock Cer (B)	6.56	7.45
BCPPSWDD	Progrem Power Swoop	5.87	6.75
BCPPTIMT	Progrem Power Timetrek (Startrek Plus)	5.87	6.75
BCPPWALL	Program Power Well (B)	4.40	5.06
BCPPWHER	Progrem Power Where (B)	5.14	5.91
BCPPWORL	Progrem Power World Geogrephy (B)	5.14	5.91
BCPPZARM	Progrem Power Zerm (B)	5.B7	6.75
BCPSSALO	Psion Seloon Selly	4.39	5.05
BSOSBEEB	Ouicksilve Beeb Art (B)	11.05	12.71
BCQSMUSI	Ouicksilve Music Processor (B)	11.05	12.71
BCRHGALA	RH Softwere Gelectic Wipeout (RHS001C)	6.61	7.60
BCRHSKIS	RH Software Ski Slelom (RHS002C)	6.61	7.60
BCSHGBLT	Simon W Hessel GB Ltd	5.13	5.90
BCSHINHE	Simon W Hessel Interitance	4.39	5.05
BCSHTRAV	Simon W Hessel Trevel Geme	4.39	5.05
BCSI3DBO	Soltwere Invasion 30 Bomb Alley	5.87	6.75
BCSIATTA	Software Invesion Atteck on Alphecenturi	5.87	6.75
BCSIEAGL	Softwere Invesion Eegles Wing (B)	5.87	6.75
BCSIGUNS	Software Invesion Gunsmoke	5.87	6.75
BCSISPOO	Softwere Invesion Spooks & Spiders (B)	5.87	6.75
BCSIVORT	Softwere Invasion Vortex	5.87	6.75
BCSMOIS	Simonsoft Oisassembler	5.87	6.75
BCSS2002	Superior Software 2002	5.67	6.75
BCSSALIO	Superior Software Alien Dropout	5.87	6.75
BCSSBATT	Superior Software Battle Tenk	5.87	B.75
BCSSCENT	Superior Software Centipede	5.87	6.75
BCSSCOLD	Superior Softwere Colditz Adventure	5.53	6.35
BCSSCRAZ	Superior Softwere Crezy Painter	5.87	6.75
BCSSFAIR	Superior Software Feirground	5.87	6.75
BCSSFROG	Superior Software Frogger	5.87	6.75
BCSSFRUI	Superior Software Fruit Mechine	5.87	6.75
BCSSGALA	Superior Softwere Galaxiens	5.87	6.75
BCSSHUNC	Superior Software Hunchbeck	5.87	6.75
BCSSINVA	Superior Software Inveders	5.87	6.75
BCSSOVER	Superior Softwere Overdrive	5.87	6.75
BCSSPONT	Superior Software Pontoon	5.13	5.90
BCSSOBER	Superior Softwere OBert	5.87	6.75
BCSSROAO	Superior Software Road Runner	5.67	6.75
BCSSSPAF	Superior Softwere Spece Fighter	5.87	6.75
BCSUJUNW	Sulis Software Junior Wordsplits	7.35	6.46
BCSUJUST	Sulis Software Just e Mot	7.35	6.46
BCSUOPEN	Sulis Softwere Open Sesame	7.35	8.46
BCSUSTAW	Sulis Softwere Sterter Wordsplits	7.35	6.46
BCSUTENF	Sulis Software Tense French	7.35	6.46
BCSUTIME	Sulis Software Time Treveller	7.35	6.46
BCSUWORO	Sulis Softwere Wordpower	7.35	8.46
BCVIOARE	Visions Oeredevil Oennis	7.00	8.05
BCVIDEMO	Visions Demoletor	7.00	8.05
BCVIPENG	Visions Pengi	7.00	8.05
BCVISNOO	Visions Snooker	6.61	7.60
BCVMDELT	Voltmece Oelte Oriver Cessette	4.39	5.05

Prices subject to variation without prior notification

TRADE ENQUIRIES WELCOME
Access & Barclaycard Accepted
Large Stocks – 24 Hour Despatch
Carriage 46p

BBC

ACORN COMPUTER

Electronequip

Title	Supplier	Description
Trafalgar	Squirrel	Ships of the line in sea battle.
Transistors Revenge	Softspot	Arcade fun in the CPU.
Trench	Virgin	Fight to save the universe.
Twin Kingdom Valley	Bugbyte	Highly complex graphic adventure.
Valley of the Kings	M.P.	Adventure game.
Vortex	S. Invasion	3D voyage into outer space with aliens and asteroids.
Wallaby	Superior	Rescue your wallaby baby from the evil monkeys.
Wheel of Fortune	Epic	Graphic adventure. 250 locations and moving chars.
Wizard	Quicksilva	Use your magic to save the princess.
Woodland Terror	M.P.	Adventure game.
Zalaga	Aardvark	Arcade action.

Suppliers of programs featured in this Guide

A&F, Unit B, Canal Side Industrial Estate, Woodbine Street East, Rochdale; **Aardvark,** 100 Ardleigh Green Road, Hornchurch, Essex; **Acornsoft,** c/o Vector Marketing, Denington Estate, Wellingborough, Northants; **Alligata,** 1 Orange Street, Sheffield; **Amcom,** 92 New Cross Street, Bradford; **ASP,** 1 Golden Square, London W1R 3AB; **Atari,** Atari House, Railway Terrace, Slough, Berks; **Bridge,** 36 Fernwood, Marple Bridge, Stockport, Cheshire; **Bug-Byte,** Mulberry House, Canning Place, Liverpool; **Chalksoft,** Lowmoor Cottage, Tonedale, Wellington, Somerset; **Comsoft,** 7 Roman Drive, Leeds, West Yorkshire; **D. Kindersley,** 1-2 Henrietta Street, London; **Dial Software,** 72 Downend Road, Downend, Bristol; **Dynabyte,** PO Box 10, Barnsley, South Yorkshire; **Epic,** 10 Gladstone Street, Kibworth, Beauchamp, Leicester; **Ganymede,** Huntsmans Walk, Rugeley, Staffs; **H&H,** 53 Holloway, Runcorn, Cheshire; **IJK,** Unit 3C, Moorfields, Moorpark Avenue, Bispham, Blackpool, Lancs; **Ixion,** 10 The Crescent, Lympsham, Weston-Super-Mare, Somerset; **Kay-Ess,** 11 Buttercup Close, Romleighs Park, Harold Wood, Essex; **Leisure Genius,** 3 Montagu Row, London W1H 1AB; **Level 9,** 229 Hughenden Road, High Wycombe, Bucks; **Melbourne,** 39 Milton Trading Estate, Abingdon, Oxon; **Microbyte,** 18 Hilgrove Road, Newquay, Cornwall; **Microdeal,** 41 Truro Road, St Austell, Cornwall; **Micro Power,** Northwood House, North Street, Leeds; **MP,** 165 Spital Road, Bromborough, Merseyside; **MRM,** 17 Cross Coates Road, Grimsby, South Humberside; **Peaksoft,** 4B Queen Street, Balderton, Newark, Notts; **Postern,** PO Box 2, Andoversford, Cheltenham, Glos; **Quicksilva,** Palmerston Park House, 13 Palmerston Road, Southampton; **Red Giant,** 3A Oakcroft Close, Pinner, Middlesex; **Salamander,** 27 Ditchling Rise, Brighton, Sussex; **Senator,** 44 Worcester Crescent, London NW7 4LL; **Serin,** PO Box 163, Slough; **Silverlind,** 156 Newton Road, Burton-On-Trent, Staffs; **Simonsoft,** 25 Tatham Road, Abingdon, Oxon; **Softspot,** 29 South Crescent, Prittlewell, Southend, Essex; **Software Comm,** Martech House, Bay Terrace, Pevensey Bay, East Sussex; **Software Invasion,** 50 Elborough Street, Southfields, London; **Squirrelsoft,** 4 Bindloss Avenue, Eccles, Manchester; **Starcade,** 2 Elworthy Avenue, Halewood Village, L26 7AA; **Superior,** Regent House, Skinner Lane, Leeds; **Virgin,** 2-4 Vernon Yard, Portobello Road, London W11 2DX; **Visions,** 1 Felgate Mews, Studland Street, London W6 9JT; **Watford Electronics,** 250 High Street, Watford, Herts.

THE MICRO USER

Games Special

A-Z guide to Games Software on the BBC Micro

Title	Supplier	Description
Adventure Quest	Level 9	Pure text adventure. 200+ locetions. Overcome umpteen obstacles guarding the Dark Tower in Middle Earth.
Air Traffic Control	Microdeal	Simulated Air Traffic Control situation where the user is responsible for take off departure of remotely controlled vehicles.
Alien Dropout	Superior Software	Arcade style shooting game where killer moths are invading earth.
Arena 3000	Microdeal	Fifteen waves of robots with which to do battle.
Artist	MRM	Computer aided design package which will eneble you to create your own pictures.
Atlantis	IJK	This includes normal Atlantis/Scramble features include depth charges, rockets, jelly fish etc.
Attack on Alpha Centauri	Software Invasion	A 3D battle for supremacy over swarms of deadly, screaming, sting dropping bug-eyed wasps.
Aviator	Acornsoft	Flight simulator.
Banana Man	MRM	Your mission is to eat 40 banana sarnies. Watch the spiders!
Base 10	Dial	How fast can you add to 10, to receive your bonus star and improve your score.
Battletank	Superior Software	A 3D tank battle game.
Beat the Bug	Bridge	A new virus is decimating mankind. You must deduce the structure of its molecule.
Beebmunch	IJK	Another version of Pacman, including ghosts, fruits, super points, screams, etc.
Beep-Beep	IJK	A version of Simon. Includes choice of number of colours and sounds.
Blitzkrieg	Software Invasior	3D tank battle with revolving, exploding tanks, 3D shell fire, etc.
Blue Dragon	MP	Adventure game.
Boris in the Underworld	Superior Software	Help Boris recover his possessions from the underworld creatures.
Brainstorm	Virgin	Unique game designed to test skill, logic, memory and nerve of two players.
Brain Teasers	Dynabyte	Six games of logic including Reversi, 3D oxo, Forty Two, Mastermind etc.
Bridgeman	Bridge	Bridge Software's version of the popular gobbler game.
Bridgemaster	Serin	Bridge tutor with world authority Terence Reese. Contains two commentary tapes, two computer tapes plus book.
Bridge to the East	Ixion	Multi screen arcade-adventure.
Bunfun	Squirrel	Frenetic assembly line happenings in a bun factory. Icing and nuts everywhere. 1-2 younger players.
Bug Bomb	Virgin	Fight off the aliens by throwing bombs at them.
Casino	IJK	Version of two popular casino games, full rules.
Castaway	Simon	A graphic adventure. Battle against giant spiders, irate chemistry teachers, etc.
Castle Assault	MRM	Scale the heavily defended castle walls to obtain your golden reward.
Castle Frankenstein	Epic	M/C adventure with over 250 locations. Find and destroy the evil Frankenstein monster.
Castle of Riddles	Acornsoft	Text adventure.

Title	Supplier	Description
Snakepit	Postern	Gobble 'em up type arcade game.
Snowball	Level9	Pure text adventure.
Space Fighter	Superior	Pilot the fighter craft against the aliens.
Space Adventure	Virgin	Graphics adventure.
Star Hawks	Key-Ess	Hold off the hovering mutant firing hawks.
Space Invaders	Bug Byte	Arcade classic.
Space Shuttle	Microdeal	Launch, park, retrieve and land, simulation.
Space Tank	Kay-Ess	Sideways scroll arcade game.
Traffic Controller	Kay-Ess	Control landing of up to 20 robot spacecraft.
Spitfire Command	Superior	Fight the battle against the Luftwaffe.
Spooks and Spiders	S. Invasion	Using stairs and platforms rescue damsel – avoiding deadly spiders.
Staircase Stampede	Comsoft	Paint the stairs, diffuse the bomb and avoid the stampede.
Star Battle	Superior	Two-player deep space dog fight.
Star Maze	S. Invasion	Discover rare jewels, dock with mothership, avoid aliens.
Star Striker	Superior	Control a three stage rocket and fight the aliens.
Star Trek	Superior	Space adventure.
Star Trek	IJK	Space adventure.
Stranded	Superior	Graphic adventure.
Stratobomber	IJK	Keep the enemy at bay and destroy the reactor.
Stockcar	Micropower	Car driving fun on the track.
Superfruit	Simonsoft	Ultimate fruit machine program.
Super Golf	Squirrel	18 hole simulation cross section view.
Superpool	S. Invasion	Real time graphic simulation.
Survivor	M.P.	Adventure game.
The Avaunting	Ixion	Adventure game with 13 magic segments.
The Cliche	Ixion	Ludicrous send up of the computer industry.
The Fallen Eagle	M.P.	Adventure game.
The Greedy Dwarf	D. Kindersley	Text adventure.
3D Maze	IJK	Battle against the clock to escape.
3D Bomb Alley	S. Invasion	Air/sea battle in 3D simulation.
3D Munchy	MRM	Pacman type game with the ghosts digging holes.
3D Space Ranger	Microbyte	Action game with on-screen ship, asteroids and trench.
3D Tank Zone	Dynabyte	Wire graphics tenk battle.
3 Deep Space	Postern	3D space action with special glasses.
Theseus	Silverlind	A million mappable 3D mazes with treasure and monsters.
2002	Superior	3D flight simulation docking to space station.

Title	Supplier	Description
Ossie	Peaksoft	Machine code arcade game.
Overdrive	Superior Software	3D car race simulation.
'Owzat	Virgin	Fight for the ashes in your own front room.
Peg Leg	IJK	Collect gold bars while fighting off pirates.
Pentiles	Silverlind	Cover a target area with a random collection of shapes.
Pengwyn	Postern	Strategy and arcade action combined.
Percy Penguin	Superior Software	Save Percee from the Snobees on the antarctic glaciers.
Pinball	MByte	Machine code simulation for up to four players.
Plank Walk	Virgin	Help Sceffolding Sid survive berserk planks.
Pole Position	Atari	3D, full colour Formula One racing car game.
Pontoon	Superior Software	Full feature version of the ever popular card game.
Pontoon & Patience	IJK	Versions of the two popular card games.
Pool	Dynabyte	Two player game using high-res graphics for accuracy and smoothness.
Proctector	Quicksilver	Protect the pods in outer space action.
Q-Man	MRM	Highly enterteining 3D pyramid game.
Q-Mans Brother	MRM	The follow up to Q-Man. Great 3D graphics.
Quest for the Holv Grail	Epic	Find and return the Holy Grail to Camelot in this machine code adventure.
Renegade Robots	Senator Software	Skillfully reassemble three cubes in a maze, avoiding the enemy.
Return to Eden	Level 9	Explore a totally alien planet in this sequel to Snowball.
Reversi	Superior Software	Board game simulation with full instructions.
Reversi	MByte	You play the micro in this machine code simulation.
Robotron 2084	Atari	Stop the robotrons and save the last humans.
Rocky	Superior Software	Drop rocks on monsters in this all action game.
Road Racer	Superior Software	A maze chase game with cars, radar and smokescreens.
Rubble Trouble	Micropower	Arcade rock pushing fun.
Sadim Castle	MP	Adventure game.
SAS Commander	Comsoft	Free the hostages and kill the terrorists.
Savage Pond	Starcade	Survive in the perils of the deep.
Scrabble 1eisute	Genius	A game of strategy involving words.
Screwball	MRM	The human corkscrew changes his surrounds avoiding the deadly black bugs.
Saa Lord	Bug Byte	Undersea fun.
Serpents Lair	Comsoft	Children's graphic adventure.
Secret Sam 1	MRM	Spy-based text adventure.
Secret Sam 2	MRM	Spy-based text adventure – more difficult.
Smash and Grab	Superior	A robber snatching gold with a PC in hot pursuit.

Title	Supplier	Description
Caterpillar	IJK	Rendition of the popular arcade game, features toadstools, spiders, fleas, etc.
Centibug	Superior Software	A game where insects attack you through an ever increasing number of mushrooms.
Champions	Peaksoft	Football management – Take your club from the 4th Division to the European Cup.
Checkout	Virgin	Change colour of squares on a grid by "running" over them twice.
Chieftain	Virgin	It's a fight to the finish in your armoured tank – fight the computer or a friend.
City Defence	Bug Byte	Need we say more.
Classic Arcade Games	Genymede	Four classic arcade games, with full colour and sound and many extra facilities. Joystick or K/B.
Colditz Adventure	Superior Software	Taunting adventure game in the perilous setting of Colditz Castle.
Colossal Adventure	Level 9	Pure text adventure. 200+ locations full scale version of the original mainframe game, "Adventure".
Community	Ixion	Strategic game where you must care for a farm, beware the famines and bank manager.
Conflict	Software Comm	Computer moderated strategy board game for two players.
Corporate Climber	Dynabyte	A race against time, the taxmen and heart failure in this arcade style game.
Cosmic Kidnap	Superior Software	Prevent the aliens from effecting the release of the captives in this fray.
Crawler	Watford Electronics	A version of the arcade geme.
Crazy Peinter	Superior Software	Help the monkey paint the squares before the hungry tribesmen catch him.
Cribbage	Superior Softwere	A version of the card game played against the computer.
Crown of Mardan	MP	Adventure game.
Cruncher	Virgin	Trample the time bombs but avoid the boots or be crunched.
Custard Pie Fight	Comsoft	Knock your opponent over with a pie. A hilarious game for one or two players.
Cybertron Mission	Micro Power	Version of the Atari game Shamus. Walk through a maze filled with robots, collecting objects.
Cylon Attack	A&F	3D space shoot out.
Dambusters	Alligator	Destroy the dam end avoid the flak.
Darts	MRM	Step up to the ocky to enjoy 501. Round the Clock and Cricket.
Deeth's Head Hole	Peaksoft	Simulation of pothole rescue call-out.
Diamond Mine	MRM	A game in which you have to guide a pipe to diamonds.
Digger	Visions	Allotment arcade capers.
Donkey Kong Jr	Atari	Help DJK rescue papa from Mario.
Draughts	Superior Software	An excellent version of the board game with some nice additions.
Draughts	Ganymede	Play the computer – no death or destruction here!
Dungeon Adventure	Level 9	Pure-text adventure. 200+ locations. Magical treasures abound in the caves of Middle Earth.

Title	Supplier	Description
Eagle's Wing	Software Invasion	Fly through a heavily guarded canyon, avoid missiles, refuel and be ready for the next confrontation.
Elite	Acornsoft	Space strategy and shooting.
Er*bert	M Byte	Help Erbert change the colour of the cubes, but avoiding unwanted guests.
Escape from Moonbase Alpha	Program Power	Graphic adventure.
Fairground	Superior Software	A simulation of a rifle range at a fun fair.
Fall of Rome	ASP	Historical strategy game.
Fantasy Adventure	Dial	Overcome the puzzles and collect to fulfil your quest.
Firienwood	MP	Adventure game.
Five A Side Socca	IJK	Two player m/c version. Joystick or keyboard.
Flags	IJK	The flags of the world are drawn in hi-res colour. Test's knowledge of flags and geography.
Flints Gold	Micrograf	Adventure game.
Frak	Aardvark	Ladders and levels game.
Franklyn's Tomb	Salamander	Adventure.
Froggy	Superior Software	You must guide Froggy across the roads and rivers to reach his home in time.
Fortress	Amcom	3D shoot out.
Fruit Machine	Superior Software	A fruit machine simulation with nudges, gambling, etc.
Fruity Freddy	Softspot	Arcade action in Mr Meany's garden.
Galaxy Birds	Superior Software	Many formations of Galaxy Birds intent on your destruction are the enemy in this game.
Gatecrasher	Quicksilver	A game of skill and strategy that will test your mind to the limits. Plus you have the opportunity to win '200.
Ghouls	Program Power	Haunted house fun.
Gideon's Gamble	Superior Software	Assist Gideon to find the treasure in this complex adventure with a nautical flavour.
Gisburne's Castle	Software Comm	Graphic arcade adventure with 3B0 different locations.
Gnasher	Superior Software	A maze chase game including power pills, ghosts and fruits.
Gunsmoke	Software Invasion	A realistic 3D Wild West gun battle.
Guy in the Hat	MRM	A game in which you have to eat "bickies" without getting eaten yourself.
Heist	Softspot	Levels and ladders in the bank.
Hobbit	Melbourne House	The well-known adventure game.
Honeybug	Silverlind	Fill the hive with honey and larvae and make the colony swarm.
Horserace	Dynabyte	Betting game for one to six players.
Horses	Kay-Ess	Choice of three arenas and six horses.
Hunchback	Superior Software	A 12 screen version of the arcade game.
Hyperdrive	IJK	Guide your laser tank around the network of passages destroying the drone aliens.

Title	Supplier	Description
Inkosi	Chalksoft	A colourful and addictive simulation with graphics and sound, ages 10 to adult.
Innerzone Shift	Ixion	Defuse enemy bombs while dodging their patrols.
Invaders	IJK	The classic game.
Invaders	Superior Software	Another version of space invaders.
Jungle Jive	Virgin	Wild animals attack you as you stroll through the jungle.
Kensington	Leisure Genius	A game of strategy involving shapes.
Killer Gorilla	Micro Power	A version of the arcade game Donkey Kong.
Kingdom of Klein	Epic	Defeat the wicked witch in this 230+ roomed adventure.
Ladder Maze	Superior Software	Find your way to the matter transmitter in this 3D maze game.
Landfall	Virgin	A space ship landing simulator.
Leap Frog	IJK	Help Froggy to get home.
Lemming Syndrome	Dynabyte	Bounce the people to safety.
Looney Lift	H&H	Operate the lift to catch rushing guests, VIPs and luggage.
Lords of Time	Level 9	Text adventure with over 200 locations.
Lost City	Superior Software	Venture into the unknown to discover the hidden mysteries of the Lost City.
Lost in Space	Salamander	Adventure.
Mazog	Red Giant	Can you reach the other side of the maze before the Mazog gets you?
Mekon Raiders	Micro Byte	3D space fight simulation.
Microbe	Virgin	Graphically stunning all-action arcade game.
Mined-Out	Quicksilver	Rescue Bill the worm from certain death. A strategy game.
MS Pac-man	Atari	Guide our heroine around tricky mazes..
Missile Base	Acornsoft	Defend your cities from the enemy.
Missile Strike	Superior Software	Defend your cities from relentless enemy missiles.
Moon Mission	Superior Software	Avoid the hazards and rescue stranded astronauts.
Mr. Wiz	Superior Software	Guide Mr. Wiz around the maze collecting fruit, avoiding gremlins
Multi-Family Games	IJK	A games compendium. Hangman, Dice, Beetle and more.
Mutant Invaders	IJK	A version of the arcade classic.
Napoleon	Mimerol	A war game where you become Bonaparte.
Neutron	Superior Software	Ride your light cycle skillfully to trap your opponent.
Nightmare Maze	MRM	Help Sleepy Joe escape the evils in his multi-screen nightmare.
Noc-A-Bloc	Virgin	Just when you thought it was safe to go back into the deep freeze.
Numberhang	Dialsoft	
Oblivion	Bug Byte	Addictive arcade action.
OG the Caveman	Simonsoft	Caveman smashes dinosaur eggs.
Old Father Time	Bug Byte	A traditional style text adventure.

SPLAT!

By MARTIN HOLLIS

SPLAT is a simple cat and mouse maze game for any age.

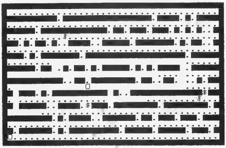
You are the star in the centre of the screen being chased by a monster – 'O' – who starts off in the top left hand corner. You have to eat all the dots, worth one point each, without being caught.

Every 100 dots you eat gives you four $ signs to eat, each worth 10 points. When your score reaches 500 another 'O' – which has been waiting in the bottom left hand corner – springs to life.

These monster 'O's do not eat the dots, but they do eat the $ signs – and you! If the two monsters hit each other, one is transported to a random position in the maze.

You have three lives to clear the screen of dots. When you lose a life you re-start from where you died.

At the beginning of the game you are asked to enter your name and to choose a level. The higher the level, the more holes there are in the walls of the maze.



At the bottom of the screen your name, score, lives left and the hi-score are displayed.

If you qualify, your name will be entered automatically in the hi-score table displayed at the end of the game.

To change the player's name, press Escape which returns you to the beginning of the game without losing the hi-score table.

PROCEDURES

PROCSCREEN	Prints screen.
PROCDOLLAR	Places $ on screen at every 100 score.
PROCSPRINKLE	Gives new dots when screen cleared.
PROCSCORES	Hiscore table.
PROCMORE	Another go option.
PROCTITLE	Prints header title.
PROCINST	Prints instruction.

Keys

A *up*	< *left*
Z *down*	> *right*

VARIABLES

C%, E%	x position of monsters.
D%, F%	y position of monsters.
H%	Level of difficulty.
H%(9)	High score.
L%	Lives.
LEFT%	Number of dots remaining on screen.
P% (40,24)	Dimension for easy access to chrs on screen.
S%	Score.
X%, Y%	Coordinates of star.
H$(9)	Table of names.

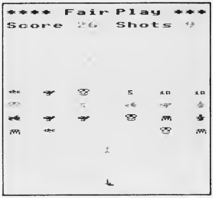
```
   10 REM **********************
   20 REM *                    *
   30 REM * SPLAT! A maze game *
   40 REM *                    *
   50 REM *  By Martin Hollis  *
   60 REM * Design - J. Barton *
   70 REM * (c)1984 Micro user *
   80 REM *                    *
   90 REM **********************
  100 ONERROR PROCMORE:GOTO200
  110 PROCTITLE
  120 PRINT''SPC(12);"By Martin Holl
is"''SPC(9);"Designed by John Barton"
'''''SPC(14);"Press SPACE"
  130 REPEAT:G=GET:UNTILG=32
  140 REPEAT
  150 PROCTITLE
  160 INPUT" What is your name",NAME$
  170 UNTILLEN(NAME$)<21
  180 DIMP%(39,22),H%(9),H$(9)
  190 FORA%=0TO9:H%(A%)=50+A%*50:H$(A
%)=CHR$(129+A%MOD4)+"Micro User.":NEXT
  200 REPEAT
  210 MODE7:VDU23;8202;0;0;0
  220 PROCINST
  230 REPEAT
  240 PROCTITLE
  250 INPUT" Enter level 1-8 (1 is ea
sy, 8 is hard)"'" then press RETURN"'
SPC(1)H%
  260 UNTILH%>0ANDH%<9:H%=(H%-1)*5
  270 FORB%=0TO39:FORA%=0TO21STEP2
  280 P%(B%,A%)=1:P%(B%,A%+1)=2:NEXT:
NEXT
  290 FORA%=1TO38:P%(A%,22)=1:NEXT
  300 FORA%=0TO22:P%(0,A%)=1:P%(39,A%
)=1:NEXT
  310 FORA%=2TO20STEP2:FORB%=0TORND(H
%)+2:P%(RND(37)+1,A%)=2:NEXT:NEXT
  320 MA%=100
  330 LEFT%=0
  340 CLS
  350 FORB%=0TO22:FORA%=0TO39
  360 IFP%(A%,B%)=0 PRINT" "; ELSEIFP
%(A%,B%)=1 PRINTCHR$(255); ELSEIFP%(A
%,B%)=2 PRINT".";:LEFT%=LEFT%+1
  370 NEXT:NEXT
  380 X%=20:Y%=11:C%=1:D%=1:E%=1:F%=2
1:S%=0:L%=3:P%(C%,D%)=0:P%(E%,F%)=0:L
EFT%=LEFT%-2:REPEAT
  390 Y%=11
  400 C%=1
  410 E%=1
  420 F%=21
  430 S%=0
  440 L%=3
  450 P%(C%,D%)=0
  460 P%(E%,F%)=0
  470 LEFT%=LEFT%-2
  480 REPEAT
  490 FORA%=0TO1
```

From Page 45

```
  500 PRINTTAB(X%,Y%);" "
  510 1FP%(X%,Y%)=2 SOUNDI,-I5,100,1:
S%=S%+1:P%(X%,Y%)=0:LEFT%=LEFT%-I
  520 IFP%(X%,Y%)=3 S%=S%+I0:FORB%=0T
O255STEP2:SOUNDI,-15,B%,0:NEXT:P%(X%,
Y%)=0
  530 IF1NKEY(-66)ANDP%(X%,Y%-I)<>1 Y
%=Y%-1ELSEIF1NKEY(-9B)ANDP%(X%,Y%+1)<
>1 Y%=Y%+1
  540 1FINKEY(-103)ANDP%(X%-1,Y%)<>1
X%=X%-IELSEIFINKEY(-104)ANDP%(X%+I,Y%
)<>I X%=X%+1
  550 FORA=0TO100:NEXT
  560 M%=0
  570 PRINTTAB(X%,Y%);"*"
  580 NEXT
  590 IFP%(C%,D%)=2 PRINTTAB(C%,D%);"
." ELSE PR1NTTAB(C%,D%);" ":P%(C%,D%)
=0
  600 IFD%<Y%ANDP%(C%,D%+1)<>1 D%=D%+
1:M%=-1
  610 1FD%>Y%ANDP%(C%,D%-I)<>1 D%=D%-
1:M%=-1
  620 IFC%>X%ANDP%(C%-1,D%)<>1ANDM%=0
 C%=C%-1 ELSEIFC%<X%ANDP%(C%+1,D%)<>I
ANDM%=0 C%=C%+1
  630 IFS%<501 GOTO670
  640 M%=0:IFP%(E%,F%)=2 PRINTTA8(E%,
F%);"." ELSE PRINTTAB(E%,F%);" ":P%(C
%,D%)=0
  650 IFF%<Y%ANDP%(E%,F%+1)<>1 F%=F%+
1:M%=-I ELSE1FF%>Y%ANDP%(E%,F%-I)<>1
F%=F%-I:M%=-1
  660 1FE%>X%ANDP%(E%-1,F%)<>IANDNOTM
% E%=E%-1 ELSEIFE%<X%ANDP%(E%+1,F%)<>
1ANDNOTM% E%=E%+1
  670 1FLEFT%<B PROCSPRINKLE
  6B0 IFC%=E%ANDD%=F% REPEAT:E%=RND(3
8)+1:F%=RND(20)+1:UNT1LP%(E%,F%)<>1
  690 PR1NTTAB(C%,D%);"O";TAB(E%,F%);
"O"
  700 PRINTTAB(0,23);"SCORE=";S%;TAB(
12,23);"L1VES=";L%;TAB(23,23);"H1GH-"
;H%(8);TAB(13,24);"P1ayer-";NAME$;
  710 1FS%>MA% MA%=MA%+I00:FORB%=0TO3
:PROCDOLLAR:NEXT
  720 IFC%=X%ANDD%=Y%ORE%=X%ANDF%=Y%
L%=L%-1:FORA%=0TO80:PRINTTAB(X%,Y%);C
HR$(RND(10)+64):SOUNDI,-15,RND(255),0
:NEXT:SOUNDO,-15,4,20:C%=I:D%=1:E%=1:
F%=21:X%=10:Y%=11:PROCSCREEN
  730 FORA=0TO100:NEXT:UNTILL%=0
  740 PROCSCORES:GOTO200
  750 DEFPROCSCREEN
  760 CLS
  770 LEFT%=0
  7B0 FORB%=0TO22:FORA%=0TO39
  790 1FP%(A%,B%)=0 VDU32: ELSEIFP%(A
%,B%)=I PRINTCHR$(255);
  B00 IFP%(A%,B%)=2 PRINT".";:LEFT%=L
EFT%+1
  BI0 IFP%(A%,B%)=3 PRINT"$";
  B20 NEXT:NEXT:ENDPROC
  B30 DEFPROCDOLLAR
  840 I%=RND(3B)
  850 J%=RND(21)
  860 PR1NTTAB(I%,J%);"$"
  870 IFP%(I%,J%)=2 LEFT%=LEFT%-1
  8B0 P%(I%,J%)=3
  890 FORK%=0TO255:SOUND1,-15,K%,0:NE
XT
  900 ENDPROC
  9I0 DEFPROCSPRINKLE
  920 H%=H%+1
  930 PROCTITLE
  940 PRINT" WelI done! You deserve a
 sprinkIing of"''''SPC(13);CHR$I29;CH
R$136"DOTS!!?!"''
  950 LEFT%=0
  960 FORB%=0TO39:FORA%=0TO21STEP2
  970 P%(B%,A%)=1
  9B0 P%(B%,A%+1)=2
  990 NEXT:NEXT
 1000 FORA%=ITO38
 1010 P%(A%,22)=1
 I020 NEXT
 I030 FORA%=0TO22
 1040 P%(0,A%)=1
 1050 P%(39,A%)=1
 1060 NEXT
 I070 FORA%=2TO20STEP2:FORB%=0TOH%+2
 10B0 P%(RND(37)+1,A%)=2
 1090 NEXT:NEXT
 I100 PROCSCREEN
 I110 X%=20:Y%=I1
 1120 C%=1:D%=1
 I130 E%=1:F%=21
 1I40 ENDPROC
 1I50 DEFPROCSCORES:IFS%<=H%(0) GOTO1
290
 I160 *FX15
 1I70 PROCTITLE
 I1B0 A%=0
 I190 REPEAT
 1200 A%=A%+1
 1210 UNTILS%<H%(A%)ORA%=9
 I220 FORB%=0TOA%-2
 I230 H%(B%)=H%(B%+I):H$(B%)=H$(B%+1)
 1240 NEXT
 I250 H%(A%-1)=S%
 1260 PRINT" CongratuIations ";NAME$'
" You are in the high score table"
 I270 H$(A%-I)=CHR$I36+NAME$
 12B0 PROCKEY
 I290 PROCTITLE
 1300 PRINTSPC(I0);CHR$(134);"   High
 Score Table"'
 1310 FORA%=8TO0STEP-1
 I320 PRINTSPC(8);H$(A%);STR1NG$(25-(
LEN(H$(A%))+LEN(STR$(H%(A%)))),".");H
%(A%)'
 1330 IFLEFT$(H$(A%),1)=CHR$I36 H$(A%
)=CHR$(128+RND(4))+MID$(H$(A%),2)
 1340 NEXT
 1350 PROCKEY
 1360 ENDPROC
 1370 DEFPROCMORE:PROCTITLE:PR1NT" An
y eore Y/N":REPEAT:G$=GET$:UNT1LG$="Y
"ORG$="N":IFG$="N" PROCTITLE:END
 I3B0 PRINT" What is your namm?":INPU
TSPC(1)NAME$:ENDPROC
 I390 DEFPROCTITLE
 1400 CLS
 14I0 PRINTCHR$(130);CHR$(157);CHR$(1
32);CHR$(141);"          S  P  L  A
T  !"';CHR$(I30);CHR$(157);CHR$(I32);
CHR$(141);"          S  P  L  A  T  !
"'
 I420 ENDPROC
 I430 DEFPROCKEY
 I440 PRINT'TAB(I4);"Press SPACE";
 I450 *FX15
 1460 REPEAT:G=GET:UNT1LG=32
 1470 ENDPROC
 14B0 DEFPROCINST
 I490 CLS
 I500 PROCTITLE
 1510 PR1NT" 1nstructions Y/N"'" ";
 1520 REPEAT:G$=GET$:UNTILG$="Y"ORG$=
"N"
 I530 IFG$="N" ENDPROC
 1540 PROCTITLE
 1550 PRINT"You start off as the ""*"
" in the middIe"'"of the screen.You m
ust eat up aII of theIittIe dots as y
ou get 1 added to your"'"score for ev
ery dot eaten. Every tiee"'"your scor
e reaches I00, 200, 300 etc."
 1560 PR1NT"you get four '$' signs wh
ich are each"'"worth I0 points when e
aten. If you eat"'"up aI1 the dots yo
u get more dots and a new maze."
 1570 PR1NT'"There is one 'O' chasing
 you round the"'"eaze, and another re
ady to be activated when your score r
eaches 500. If the two 'Os' eeet one
of them is sent to anotherpart of the
 eaze."
 1580 PRINT'"You also get a personali
sed High Score"'"TabIe for nine score
s. If you qualify"'"your naee wilI be
 entered autoeatically."
 1590 PROCKEY
 I600 CLS
 1610 PROCTITLE
 I620 PR1NT''TAB(7);CHR$(136);CHR$(13
4);"Keys for the GAME."''TAB(11);CHR$
(131);"A   =   Up."''TAB(11);CHR$(132
);"Z   =   Down."''TAB(11);CHR$(133);
"<   =   Left."''TAB(11);CHR$(130);">
   =   Right."'''
 1630 PROCKEY
 I640 ENDPROC
```

FAIR PLAY

By D.H. IBBOTSON


**** Fair Play ***
Score 26 Shots 9

THE fairground has come to your micro in the form of Happy Harry's shooting gallery. All you have to do is hit the targets with the darts in order to win one of Harry's fantastic prizes. (If the prizes are not quite up to your expectations then simply change the data on line 160.)

Hitting a number will increase your darts. Hitting a face will make all the targets retreat. Hitting anything else will increase your score.

Clear all the targets and a new set will appear in new colours, sometimes making the targets or gun partly invisible. So try to hit the numbers first, followed by everything except the faces. Shoot the faces last of all in order to progress to the next card.

Instructions are contained in the program. Only the space bar is strictly necessary in order to play the game. I hope you enjoy the Fairground organ music.

The data on line 460 may be changed if the score, or the number of extra darts, is not to your liking.

The formula is simple, 0 to 9 is added to the score, 10 or greater will be added to the darts, and –2 (use –2 only) will force a retreat.

Likewise characters 224 to 232 can be redefined if you dislike the targets.

ARRAYS

A$( )	Targets.
C$( )	Colours in the form of strings.
D$(1)	Dart / D$(0) delete dart.
prize$( )	All the prizes.
R%( )	Temporary stores for random numbers.
S%( )	Scores for the various targets.
s%( )	Old scores for score ladder.
s$( )	Old names for score ladder.

VARIABLES

A%	Permanently set 165.
a%	Permanently set to OSBYTE address &FFF4.
A$	Removes bottom line of targets when they retreat.
B%	X axis position of the gun.
C%	Master counter.
D%	Darts flag 1 if fired, 0 if not fired.
G$	The gun.
H%	Height of targets.
I%	Jump out of loop flag.
J%	Counts hits.
L%	Level or card.
M%	Time value for each note.
N%	Pitch value for each note.
O%	Removes bottom line of targets if set flag.
S%	Total shots remaining.
s%	Old total shots remaining.
T%	Total scored.
t%	Old total scored.
X%	Darts X axis.
Y%	Darts Y axis.

PROCEDURES

PROCmulti_colour	Displays Mode 6 (normally a single colour mode) in multi-colour by using *FX19 and then rapidly changing the palette in order to display bands of colour.
PROCsetup_characters	Sets the string array A$( ) to random characters or targets.
PROCcheck (A%)	Finds out which target (A%) has been hit and then pays out accordingly.
PROCchange	Selects fresh colour from the palette.
FNp (A$)	PEEKs the screen more or less as described on page 432 of the User Guide. If a hit is detected removes it from the appropriate string, then calls PROC-change.
FNs	Gets the name of the player for use in PROCscore_ladder.
FNT	Resets data pointer to line 1180 if the end of the music, as indicated by M%=0, has been reached.

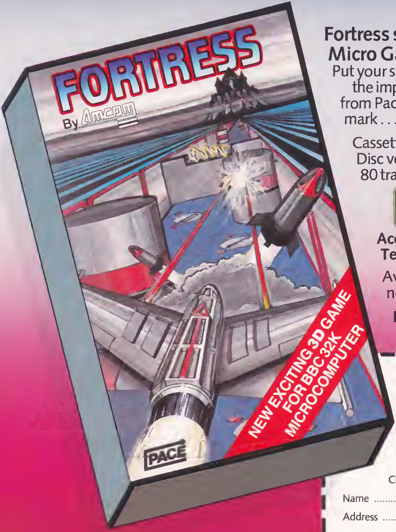
```
   10 REM **** Fair play ****
   20 REM (c) The Micro User
   30 REM Program by D.H.Ibbotson.
   40 REM  Music by T.Howes.
   50 ENVELOPE1,4,90,-15,-15,10,20,20
,126,0,0,-126,126,126
   60 ENVELOPE2,4,-1,1,0,20,20,0,126,
0,0,-126,126,126
   70 ENVELOPE3,3,0,0,0,0,0,0,121,-10
,-5,-2,-120,-120
   80 ENVELOPE4,0,0,0,0,0,0,0,126,-4,
-3,-3,126,100
   90 ENVELOPE5,10,1,-2,1,10,10,10,70
,-10,-8,-5,-50,0
  100 DIMA$(3),C$(3),D$(1),R%(3),S%(7
)
  110 REM arrays for score ladder
  120 DIMs%(8),s$(8):FORA%=0TO8:s%(A%
)=0:s$(A%)="Happy Harry  ":NEXT
  130 REM arrays for prizes
  140 DIMprize$(7)
  150 RESTORE:FORa=0TO7:READprize$(a)
:NEXT
  160 DATA "a plastic ring","a cardbo
ard dolls house","a fish in a bag","a
 coconut","a rag doll","a plaster duc
k","a toy whistle","any prize on the
stall"
  170 MODE6
  180 VDU23;8202;0;0;0;
  190 PRINTTAB(9,2)"Welcome to the MI
CRO fair"'''
  200 PRINTSPC2"try your luck at the
shooting gallery"'''
  210 PRINTSPC11"A prize every time !
!"'''
  220 PRINTSPC2"everyone wins on  Hap
py Harry's pitch"''
  230 PRINTSPC13"5  darts per go"''
  240 PRINTSPC14"PRESS ANY KEY"
  250 PROCmulti colour:FORa=0TO1000:N
EXT
  260 PRINTTAB(14,2)"INSTRUCTIONS"''
  270 PRINTSPC9"Z move gun to the rig
```

→

From Page 47

```
ht"'''
  280 PRINTSPC9"X move gun  to the le
ft"'''
  290 PRINTSPC9"SPACE  to fire  a  da
rt"'''
  300 PRINTSPC14"PRESS ANY KEY"
  310 PROCmulti_colour
  320 MODE5:VDU19.1,1;0:
  330 REM define characters
  340 VDU23,224,0,6,140,127,255,206,1
03,0
  350 VDU23.225,0,0,59,223,254,60,8,5
6
  360 VDU23,226,24,24,24,126,44,124,2
8,56
  370 VDU23,227,0,16,123,222,115,8,0,
0
  380 VDU23.228,0,0,62,107,62.85,85,6
5
  390 VDU23,229,0,0,56,32,48,8,56,0
  400 VDU23,230,0,0,0,78,202,74,238,0
  410 VDU23,231,126,153,153,102,70,40
,66,60
  420 VDU23,232,16,56,16,16,16,16,56,
40
  430 VDU23,233,16,16,16,24,20,60,31,
15
  440 REM set up score array
  450 RESTORE460:FORa=0TO7:READS%(a):
NEXT
  460 DATA 2,1,3,4,5,10,20,-2
  470 PROCsetup characters
  480 REM the gun
  490 G$=C$(2)+" "+CHR$(233)+" "
  500 REM the darts
  510 D$(0)=" "+CHR$(8)+CHR$(10)+" ":
D$(1)=C$(1)+CHR$(232)+CHR$(8)+CHR$(10
)+" "
  520 A$="                    "
  530 VDU19,3,10;0;
  540 VDU23;8202;0;0;0;
  550 *FX9 100
  560 *FX10 100
  570 PRINT"**** Fair  Play ****"'C$(
2)"Score      Shots"
  580 REM constants
  590 A%=135:a%=&FFF4
  600 REM U% V% & Z% temp numbers
  610 REM U$ V$ & Z$ temp strings
  620 REM initial settings for variab
les
  630 T%=0:t%=1:S%=5:s%=1:L%=0
  640 B%=9
  650 TIME=1:M%=0:N%=0:RESTORE1180
  660 REPEATJ%=24
  670 C%=0:D%=0:H%=16:X%=0:Y%=4
  680 I%=0
  690 O%=0
  700 REM main loop
  710 REPEATC%=C%+1AND7:IF TIME>=M%*1
0READN%,M%:IFFNT SOUND&12,5,(N%-17)*4
,255:SOUND&13,4,(N%-25)*4,255:TIME=0
  720 PRINTTAB(B%,30)G$:VDU31,X%,Y%:P
RINT;FNp(D$(D%)):IFO%O%=0:PRINTTAB(0,
H%+6);A$:H%=H%-2:IFH%=4H%=6
  730 IF(C%AND1)PRINTTAB(0,H%+(C%AND1
4))C$(C%MOD3+1)A$(C%DIV2)ELSEA$(C%DIV
2)=FNmove(A$(C%DIV2),C%AND3)
  740 IFY%<>4Y%=Y%-1:IFY%=3Y%=4:D%=0
  750 IFD%=0IFINKEY-99D%=1:X%=B%+1:Y%
=29:S%=S%-1:SOUND&10,-15,5,8
  760 IFB%>1IFINKEY-98B%=B%-1
  770 IFB%<16IFINKEY-67B%=B%+1
  780 IFs%s%=0:PRINTTAB(17,2)C$(1);S%
" ":IFS%<1I%=1 ELSEIFt%<>T%PRINTTAB(6
,2)C$(1);T%
  790 UNTILI%:IFS%>0FORu%=100TO200STE
P3:SOUND1,1,u%,3:NEXT:L%=L%+1:J%=24:I
%=0:S%=5:s%=1:PROCsetup_characters:PR
OCchange:RESTORE1180:UNTIL0
  800 UNTIL1:Z$=prize$(7):IFT%<699Z$=
prize$((T%DIV100)AND7)
  810 *FX15
  820 FORA%=150TO0STEP-1:SOUND1,-A%DI
V10,A%,1:NEXT
  830 MODE6:PRINTTAB(0,2)"Well done y
ou have won"''''Z$''''"PRESS ANY KEY
TO ENTER YOUR NAME"''''"If your score
 is high enough,of course":VDU23;8202
;0;0;0;
  840 PROCmulti_colour:PROCscore_ladd
er(T%):GOTO170
  850 REM end main loop
  860 DEFFNmove(A$,A%)IFA%Z$=LEFT$(A$
,1):=RIGHT$(A$,19)+Z$ ELSEZ$=RIGHT$(A
$,1):=Z$+LEFT$(A$,19)
  870 DEFFNp(A$)Z%=USRa%:IF?&469<>32D
%=0:IFY%>H%-1U%=(Y%-H%)DIV2:A$(U%)=LE
FT$(A$(U%),X%-1)+"   "+RIGHT$(A$(U%),
18-X%):Y%=4:s%=1:PROCcheck(S%(?&469-1
28)):J%=J%-1:IFJ%=0:I%=1
  880 IFY%=H%s%=1
  890 =D$(D%)
  900 REM checks on characters hit th
en pays out accordingly
  910 DEFPROCcheck(A%)IFA%>9S%=S%+A%D
IV2:SOUND&11,1,100,12:ENDPROC
  920 IFA%=-2O%=1:SOUND&11,2,120,15:E
NDPROC
  930 T%=T%+A%+(3-U%)*10:SOUND&11,3,1
00,4:ENDPROC
  940 DEFPROCsetup_characters:FORa=0T
O3:C$(a)=CHR$(17)+CHR$(a):A$(a)="":FO
Rz=0TO6:A$(a)=A$(a)+"  "+CHR$(223+RND
(8)):A$(a)=LEFT$(A$(a),20):NEXT,:ENDP
ROC
  950 REM adds multi colour to mode 6
  960 DEFPROCmulti_colour
  970 REM do not abreviate or add to
lines 990 & 1000
  980 *FX15
  990 REPEAT*FX19
 1000 FORA%=1TO6:VDU19,1,A%;0;::NEXT
:UNTILINKEY(0)<>-1:CLS:ENDPROC
 1010 REM change colours
 1020 DEFPROCchange R%(1)=RND(15):REP
EATR%(2)=RND(15):R%(3)=RND(15):UNTILR
%(1)<>R%(2)ANDR%(2)<>R%(3)ANDR%(1)<>R
%(3):FORa=1TO3:VDU19,a,R%(a);0;:NEXT:
ENDPROC
 1030 REM score ladder
 1040 DEFPROCscore_ladder(S%):*FX15
 1050 VDU4,23;8202;0;0;0;:IFS%>s%(7)P
RINTTAB(11,10)"Please enter"''SPC11"Y
our    name"'':IFFNs A%=-1:REPEATA%=A
%+1:UNTILS%>s%(A%):FORB%=7TOA%STEP-1:
s%(B%+1)=s%(B%):s$(B%+1)=s$(B%):NEXT:
s%(A%)=S%:s$(A%)=Z$
 1060 B$="-------------"
 1070 CLS:VDU23;8202;0;0;0;:PRINTSPC7
"Harry's Hall of HEROES"''SPC11" Last
 Score=";S%''"   Scores  "+B$+"  Name
s"':FORA%=0TO7:PRINTSPC4" ";s%(A%)TAB
(11)B$TAB(25)" "s$(A%)':NEXT:PRINTSPC
11"PRESS ANY KEY":*FX15
 1080 VDU19,1,6;0;:REPEATUNTILGET:END
PROC
 1090 DEFFNs A%=0:Z$=""
 1100 VDU7
 1110 X%=GET:IFX%=13=1
 1120 IFX%=127ANDA%=0GOTO1100
 1130 IFX%=127:A%=A%-1:Z$=LEFT$(Z$,A%
)ELSEIFA%<12A%=A%+1:Z$=Z$+CHR$X%ELSEV
DU7
 1140 PRINTTAB(11,14)" "Z$" ":GOTO111
0
 1150 DEFFNT IFM%=0RESTORE1180:=0ELSE
=1
 1160 UNTIL0
 1170 REM mid C=25
 1180 DATA 37,6,27,6,35,6,25,6
 1190 DATA 34,2,33,2,34,2,37,4,36,2,3
5,2,34,2,33,2,34,4,31,2
 1200 DATA 32,2,31,2,32,2,35,4,34,2,2
5,2,30,2,34,2,37,4,38,2
 1210 DATA 39,2,38,2,39,2,41,4,41,2,4
1,6,39,4,38,2
 1220 DATA 37,6,27,6,35,6,25,6
 1230 DATA 34,2,33,2,34,2,37,4,36,2,3
5,2,34,2,31,2,32,6
 1240 DATA 33,6,41,4,39,1,41,1,42,6,3
4,4,35,2
 1250 DATA 36,2,39,2,37,2,35,4,29,2,3
0,6,34,2,35,2,36,2
 1260 DATA 0,0
```

Once you've got FORTRESS you might never see the end of it!



Fortress sets the pace for other Micro Games

Put your skills to the ultimate test and overcome the impossible with the explosive new game from Pace. Meet the challenge and make your mark . . . **The fortress awaits you.**

Cassette **£8.85** inc. VAT
Disc version available. Usable on both 40 and 80 track drives. **£11.95** inc. VAT.

Access/Barclaycard orders.
Tel: (0274) 729306

Available direct from Pace or from your nearest Micro Dealer.

PACE, 92 New Cross Street,
Bradford, BD5 8BS.

Please send me □ 'Fortress' Cassettes at **£8.95** inc VAT
+55p P+P TOTAL £9.50
And/or □ 'Fortress' Discs at **£11.95** inc. VAT
+55p P+P. TOTAL £12.50.
Total remittance enclosed ....
Access/Barclaycard No ....
Cheque/PO No ....
Name ....
Address ....
....
.... Tel .... PACE

A realistic interpretation of the gentle art of playing bowls by ALAN TOPHAM

CROWN GREEN

CROWN Green is a vivid simulation of the game for two players. The program allows for different weights of bowl, 16 bowling directions and left or right bias. The wood – the name remains, even though many bowls are now made of plastic – even slows down to a halt realistically.

For those of you who like to play rough, full allowance is made for cannoning of woods into one another.

Each player plays two woods alternately from the footer (a non-slippable mat). The player who bowls the jack also retains possession of the footer until his bowl has stopped running.

If the jack comes to rest within less than 21 yards – strength 150 in our game – or is bowled directly off the green, the player loses it to the opponent.

If it is knocked off the green during an end, that end is replayed and the same player retains the jack.

At the completion of each end a player scores one chalk – point – if his bowl is nearer the jack and two points if both are. The first player to 21 points is the winner.

Before each delivery instructions are given in the boxes at the top and bottom of the screen and a hint is given as to which way the run of the bowl will bend.

The lines drawn between each bowl and the jack at the finish of each end represent measuring strings and the player who wins the end bowls the jack at the start of the next one.

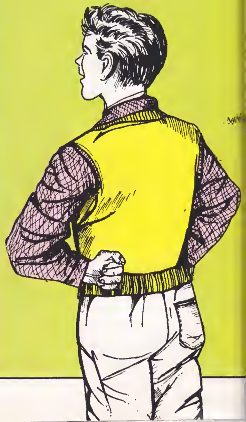
It is sometimes better to alter the direction and peg – bias – to avoid knocking your opponent in or yourself out.

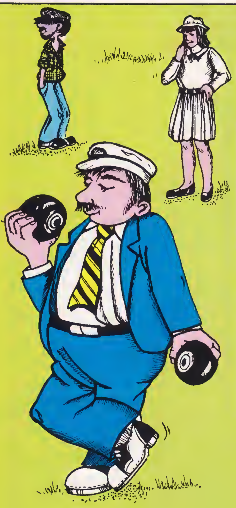
VARIABLES

A%,B%	Coordinates of the running bowl.
C%	Colour of each bowl.
P%	TRUE if a bowl has been hit or 2 if it has gone off the green.
M%,N%	Coordinates of any hit bowl.
H%	Number of bowl being bowled.
A%	Player.
HIT%	TRUE if any bowl has been hit.
S%,T%	Determines direction of running bowl and causes it to curve (bias).
F%	TRUE if jack is being bowled.
X%,Y%	Position of the footer.
A$,B%,C%	PROCmessage parameters.
D%	Direction of bowl pointer arrow.
G$	Bias selection (finger or thumb peg).
E%	Strength factor.
MANX%,MANY%	Position of man at start of delivery.
MANX2,MANY2	Position of man at the footer.
L%	Count for running bowl – random factor at line 990 brings an element of chance.
G%	1 if the jack has been knocked off during an end.

PROCEDURES

PROCbowl	Moves current bowl and calls PROCwhbowlA and tests for a hit.
PROCover	The finish of each end.
PROCbend and PROCacross	Causes the bowl to bend according to the bias (G$).
PROCmove	Moves only the "hit" bowls.
PROCwhbowlA	Tests for a hit on any still bowl.
PROCscores	Measures the bowls from the jack.
PROCslow	Ensures that each bowl wobbles to a halt instead of stopping dead.



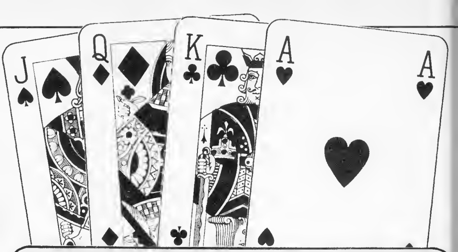


```
   10 REM CROWN GREEN BOWLS
   20 REM BY ALAN TOPHAM
   30 REM THE MICRO USER
   40 ON ERROR GOTO2490
   50 DIM A%(5),B%(5),C%(5),P%(5),M%(
5),N%(5)
   60 PROCinit:MODE1
   70 VDU19,1,4,0,0,0,19,3,2,0,0,0
   80 COLOUR1:PRINTTAB(1,1);CHR$248:P
RINTTAB(38,30);CHR$248:COLOUR2:PRINTT
AB(38,1);CHR$248:PRINTTAB(1,30);CHR$2
48
   90 PROCscreen:PROCprintscores
  100 REPEAT
  110 C%(H%)=A%
  120 IF A%=2 PROCmessage("BLUE BOWL
TO PLAY",11,30) ELSE PROCmessage("YEL
LOW BOWL TO PLAY",10,30)
  130 PROCdirection
  140 PROCbias
  150 PROCstrengh
  160 PROCposman
  170 PROCbowl
  180 IF H%=6 THEN PROCover
  190 UNTIL score1%>=21 OR score2%>=2
1
  200 PROCfin
  210 END
  220 DEFPROCinit
  230 VDU23,240,3,3,3,1,1,7,15,27,23,
241,192,192,192,128,128,224,224,224,2
3,242,51,99,195,131,3,3,3,3,23,243,22
4,224,240,216,204,198,195,193,23,244,
2,2,2,2,2,2,2,2,23,245,64,64,64,64,64
,64,64,64,23,246,0,56,124,124,124,56,
0,0
```

From Page 51

```
  240 VDU23,241,192,192,192,128,128,2
24,224,224
  250 VDU23,224,16,56,84,146,16,16,16
,16,23,225,124,10,10,17,16,32,32,64,2
3,226,31,3,5,9,17,32,64,128,23,227,8,
6,1,7,25,97,129,0,23,228,8,4,2,255,2,
4,8,0,23,229,0,129,97,25,7,1,6,8,23,2
30,128,64,32,17,9,5,3,31
  260 VDU23,231,64,32,32,16,17,10,10,
124,23,232,8,8,8,8,73,42,28,8,23,233,
2,4,4,8,136,80,80,62,23,234,1,2,4,136
,144,160,192,248,23,235,0,129,134,152
,224,128,96,8,23,236,16,32,64,255,64,
32,16,0,23,237,8,96,128,224,152,134,1
29,0
  270 VDU23,238,240,192,160,144,136,4
,2,1,23,239,62,80,80,136,8,4,4,2,23,2
47,0,0,24,60,60,24,0,0,23,248,16,48,1
12,240,16,16,16,16
  280 HIT%=FALSE:Z=0:F%=TRUE:H%=1:sco
re1%=0:score2%=0:A%=2:X%=650:Y%=200
  290 ENDPROC
  300 DEFPROCscreen
  310 MOVE20,20:DRAW1259,20:DRAW1259,
1003:DRAW20,1003:DRAW20,20
  320 VDU24,75;75;1204;948;
  330 GCOL0,131:CLG
  340 ENDPROC
  350 DEFPROCmessage(A$,B%,C%)
  360 VDU4
  370 PRINTTAB(B%,C%);A$
  380 VDU5
  390 ENDPROC
  400 DEFPROCdirection
  410 *FX15,1
  420 PROCmessage("SELECT DIRECTION W
ITH (Z&X) KEYS",4,1)
  430 GCOL0,0:D%=224
  440 MOVEX%,Y%:VDU42,11,8,D%
  450 REPEAT
  460 E=INKEY(500)
  470 IF E=88 THEN D%=D%+1:IF D%=240
THEN D%=224
  480 IF E=90 THEN D%=D%-1:IF D%=223
THEN D%=239
  490 VDU127,D%
  500 UNTIL E=&0D
  510 ENDPROC
  520 DEFPROCbias
  530 GOSUB2350
  540 IF D%=227 OR D%=228 OR D%=229 P
ROCmessage("SELECT BIAS (Z UP X DOWN)
",7,1) ELSE IF D%=235 OR D%=236 OR D%
=237 PROCmessage("SELECT BIAS (X UP Z
 DOWN)",7,1) ELSE PROCmessage("SELECT
 BIAS (Z LEFT X RIGHT)",7,1)
  550 REPEAT:G$=GET$:UNTIL G$="Z" OR
G$="X"
  560 GOSUB2350
  570 ENDPROC
  580 DEFPROCstrengh
  590 PROCmessage("STRENGTH (MAX 300)
"   ,7,1)
  600 VDU4
  610 INPUTTAB(25,1),E%:IF E%>300 THE
N GOSUB2350:PROCstrengh
  620 GOSUB2340
  630 GOSUB2350
  640 ENDPROC
  650 DEFPROCposman
  660 MOVEX%,Y%:VDU9,127,11,9,127
  670 RESTORE 710
  680 FOR Q=224 TO D%
  690 READ MANX%,MANY%,MANX2,MANY2,S%
,T%
  700 NEXT
  710 DATA-57,-20,-57,60,0,5,-77,-20,
-57,60,2,5,-96,-20,-57,60,4,5,-96,38,
-57,78,5,2,-96,96,0,96,5,0,-96,108,0,
75,5,-2,-96,120,24,55,4,-5,-38,120,24
,55,2,-5
  720 DATA20,120,24,55,0,-5,40,120,24
,55,-2,-5,72,120,24,55,-4,-5,72,80,24
,48,-5,-2,72,40,0,40,-5,0,72,10,0,50,
-5,2,72,-20,-57,60,-4,5,10,-20,-57,60
,-2,5
  730 MOVEX%+MANX%,Y%+MANY%:GOSUB2360
  740 IF D%=224 OR D%=225 OR D%=226 O
R D%=227 OR D%=236 OR D%=237 OR D%=23
8 OR D%=239 MOVE X%+MANX%+60,Y%+MANY%
-60 ELSE MOVE X%+MANX%-24,Y%+MANY%-48
  750 IF F% GCOL3,3:VDU247 ELSE GCOL3
,A%:VDU246
  760 TIME=0:REPEAT UNTILTIME>90
  770 IF F% VDU8,247 ELSE VDU8,246
  780 MOVE X%+MANX%,Y%+MANY%:GOSUB236
0
  790 MOVE X%+MANX2,Y%+MANY2:GOSUB236
0
  800 ENDPROC
  810 DEFPROCbowl
  820 IF H%=1 THEN GCOL3,3 ELSE GCOL3
,A%
  830 A%(H%)=X%:B%(H%)=Y%
  840 MOVE A%(H%),B%(H%):IF H%=1 VDU2
47 ELSE VDU246
  850 L%=0
  860 REPEAT
  870 L%=L%+1
  880 IF L% MOD 50=0 PROCbend
  890 PROCmove
  900 IF H%=1 THEN GCOL3,3 ELSE GCOL3
,C%(H%)
  910 X1%=A%(H%)+S%:Y1%=B%(H%)+T%
  920 MOVE A%(H%),B%(H%):IF H%=1 VDU2
47 ELSE VDU246
  930 MOVE X1%,Y1%:IF H%=1 VDU247 ELS
E VDU246
  940 A%(H%)=X1%:B%(H%)=Y1%
  950 IF A%(H%)<90 OR A%(H%)>1165 OR
B%(H%)>940 OR B%(H%)<110 THEN SOUND0,
-15,6,1:MOVEA%(H%),B%(H%):GOSUB2370:P
%(H%)=2:L%=E%
  960 IF L%<60 THEN990
  970 PROCwhbowlA:IF HIT% SOUND0,-12,
3,1
  980 PROCmove
  990 IF RND(4)=2 L%=L%+2
 1000 IF L%>(E%-60) THEN PROCslow
 1010 UNTIL L%>=E%
 1020 IF P%(1)=2 THEN PROCjackoff:END
PROC
 1030 H%=H%+1
 1040 REPEAT
 1050 HIT%=FALSE
 1060 FOR F=1 TO 5
 1070 IF P%(F)=2 THEN 1090
 1080 P%(F)=FALSE
 1090 NEXT
 1100 PROCwhbowlA:PROCmove
 1110 IF P%(1)=2 G%=1:PROCjackoff:HIT
%=FALSE
 1120 UNTIL HIT%=FALSE
 1130 IF F%=TRUE AND E%<150 THEN PROC
message("FAILED TO SET A MARK(minimum
 150)",4,1):SOUND1,-15,1,3:T=TIME:REP
EAT UNTIL TIME-T>200:CLG:F%=TRUE:H%=1
:GOTO1160
 1140 MOVEX%+MANX2,Y%+MANY2:GOSUB2360
 1150 IF F%=TRUE THEN F%=FALSE:ENDPRO
C
 1160 PROCchplayer
 1170 ENDPROC
 1180 DEFPROCover
 1190 MOVEX%-120,Y%:GOSUB2360:PROCchp
layer:MOVEX%+80,Y%+20:GOSUB2360
 1200 PROCmessage("MEASURING NOW",12,
1)
 1210 X%=A%(1):Y%=B%(1):IF X%<200 THE
N X%=200 ELSE IF X%>1030 X%=1030
 1220 IF Y%<200 Y%=200 ELSE IF Y%>800
 Y%=800
 1230 PROCscores
 1240 PROCprintscores
 1250 FOR T=1 TO 3000:NEXT
 1260 GOSUB2350
 1270 IF score1%>=21 OR score2%>=21 T
HEN ENDPROC
 1280 PROCmessage("PRESS SPACE BAR",1
2,30)
 1290 FOR T=1 TO 5:P%(T)=0:NEXT
 1300 G=GET:IF G<>32 THEN 1300
 1310 CLG:H%=1:F%=TRUE
 1320 ENDPROC
 1330 DEFPROCbend
 1340 IF D%=227 OR D%=228 OR D%=229 O
R D%=235 OR D%=236 OR D%=237 PROCacro
ss:ENDPROC
 1350 IF T%>1 T%=T%-1:IF T%<=1 T%=1 E
LSE IF T%<-1 T%=T%+1:IF T%>=-1 T%=-1
 1360 IF G$="Z" S%=S%-1
 1370 IF G$="X" S%=S%+1
 1380 ENDPROC
 1390 DEFPROCacross
 1400 IF T%>=4 T%=4 ELSE IF T%<=-4 T%
=-4
 1410 IF S%>1 S%=S%-1:IF S%<=1 S%=1 E
LSE IF S%<-1 S%=S%+1:IF S%>=-1 S%=-1
 1420 IF D%=235 OR D%=236 OR D%=237 T
HEN 1460
 1430 IF  G$="Z" T%=T%+1:S%=S%+1
 1440 IF G$="X" T%=T%-1:S%=S%+1
 1450 ENDPROC
 1460 IF G$="Z" T%=T%-1:S%=S%-1
 1470 IF G$="X" T%=T%+1:S%=S%-1
 1480 ENDPROC
 1490 DEFPROCmove
 1500 FOR W%=1 TO H%-1
 1510 IF P%(W%)=2 OR P%(W%)=FALSE THE
N 1590
 1520 IF W%=1 THEN GCOL3,3 ELSE GCOL3
,C%(W%)
 1530 X1%=A%(W%)+M%(W%):Y1%=B%(W%)+N%
(W%)
 1540 MOVE X1%,Y1%:IF W%=1 VDU247 ELS
E VDU246
 1550 MOVE A%(W%),B%(W%):IF W%=1 VDU2
47 ELSE VDU246
 1560 A%(W%)=X1%:B%(W%)=Y1%
 1570 L%=L%+1
 1580 IF A%(W%)<90 OR A%(W%)>1165 OR
B%(W%)>940 OR B%(W%)<110 THEN SOUND0,
-15,6,1:MOVEA%(W%),B%(W%):GOSUB2370:P
%(W%)=2:IF W%=1 L%=E%:G%=1
 1590 NEXT
 1600 ENDPROC
 1610 DEFPROCscores
 1620 FOR Y=2 TO 5
 1630 IF P%(Y)=2 THEN 1820
 1640 A=0
 1650 X1=A%(Y):Y1=B%(Y)
 1660 REPEAT
 1670 IF X1<A%(1) THEN X1=X1+1 ELSE X
1=X1-1
 1680 IF Y1<B%(1) THEN Y1=Y1+1 ELSE Y
1=Y1-1
 1690 A=A+1
 1700 UNTIL X1=A%(1) OR Y1=B%(1)
 1710 M%(Y)=A*4
 1720 A=0
 1730 REPEAT
 1740 IF X1<A%(1) THEN X1=X1+1
 1750 IF X1>A%(1) THEN X1=X1-1
 1760 IF Y1<B%(1) THEN Y1=Y1+1
 1770 IF Y1>B%(1) THEN Y1=Y1-1
 1780 A=A+1
 1790 UNTIL X1=A%(1) AND Y1=B%(1)
 1800 N%(Y)=A*3
 1810 MOVEA%(1)+12,B%(1)-12:GCOL3,C%(
Y):DRAWA%(Y)+12,B%(Y)-12
 1820 NEXT
 1830 FOR Y=2 TO 5:IF P%(Y)=2 THEN 18
60
 1840 MOVEA%(1)+12,B%(1)-12:GCOL3,C%(
Y):DRAWA%(Y)+12,B%(Y)-12
 1850 FOR T=1 TO 1000:NEXT
 1860 NEXT
 1870 FOR Y=2 TO 5:IF P%(Y)=2 THEN P%
(Y)=5000:M%(Y)=Y:NEXT ELSE P%(Y)=(M%(
Y)+N%(Y)):M%(Y)=Y:NEXT
 1880 FOR X=1 TO 3
 1890 FOR Y=2 TO 4
 1900 IF P%(Y)>P%(Y+1) THEN P%(1)=P%(
Y):P%(Y)=P%(Y+1):P%(Y+1)=P%(1):M%(1)=
M%(Y):M%(Y)=M%(Y+1):M%(Y+1)=M%(1)
 1910 NEXT
 1920 NEXT
 1930 IF P%(2)=5000 THEN ENDPROC
 1940 IF C%(M%(2))=2 THEN PROCblwin E
LSE PROCylwin
 1950 ENDPROC
 1960 DEFPROCprintscores
 1970 FOR A=80 TO 120 STEP 10:SOUND1,
-11,A,1:NEXT
 1980 VDU4
 1990 COLOUR1:PRINTTAB(3,30);score1%:
COLOUR2:PRINTTAB(35,30);score2%
 2000 VDU5
 2010 ENDPROC
 2020 DEFPROCjackoff
 2030 PROCmessage("   JACK OFF THE GR
EEN ",7,1):SOUND1,-15,1,3:T=TIME:REPE
AT UNTIL TIME-T>200
 2040 FOR T=1 TO 5:P%(T)=FALSE:NEXT
 2050 H%=1:CLG:F%=TRUE
 2060 IF G%=1 A%=C%(2) ELSE PROCchpla
yer
 2070 G%=0
 2080 ENDPROC
 2090 DEFPROCwhbowlA
 2100 HIT%=FALSE
 2110 FOR Q%=1 TO H%-1
 2120 IF P%(Q%)=2 OR P%(Q%)=TRUE THEN
 2210
 2130 IFPOINT(A%(Q%)-4,B%(Q%)+3)<>3 T
HEN M%(Q%)=2:N%(Q%)=-3:P%(Q%)=TRUE:HI
T%=TRUE
 2140 IFPOINT(A%(Q%)+12,B%(Q%)+6)<>3
THEN M%(Q%)=0:N%(Q%)=-3:P%(Q%)=TRUE:H
IT%=TRUE
 2150 IFPOINT(A%(Q%)+28,B%(Q%)+3)<>3
THEN M%(Q%)=-2:N%(Q%)=-3:P%(Q%)=TRUE:
HIT%=TRUE
 2160 IFPOINT(A%(Q%)-6,B%(Q%)-12)<>3
THEN M%(Q%)=3:N%(Q%)=0:P%(Q%)=TRUE:HI
T%=TRUE
 2170 IFPOINT(A%(Q%)+30,B%(Q%)-12)<>3
 THEN M%(Q%)=-3:N%(Q%)=0:P%(Q%)=TRUE:
HIT%=TRUE
 2180 IFPOINT(A%(Q%)+28,B%(Q%)-28)<>3
 THEN M%(Q%)=-2:N%(Q%)=3:P%(Q%)=TRUE:
HIT%=TRUE
 2190 IFPOINT(A%(Q%)+12,B%(Q%)-30)<>3
 THEN M%(Q%)=0:N%(Q%)=3:P%(Q%)=TRUE:H
IT%=TRUE
 2200 IFPOINT(A%(Q%)-4,B%(Q%)-28)<>3
THEN M%(Q%)=2:N%(Q%)=3:P%(Q%)=TRUE:HI
T%=TRUE
 2210 NEXT
 2220 ENDPROC
 2230 DEFPROCblwin
 2240 IF C%(M%(2))=C%(M%(3)) THEN PRO
Cmessage("TWO TO BLUE",14,30):score1%
=score1%+2 ELSE PROCmessage("ONE TO B
LUE",14,30):score1%=score1%+1
 2250 A%=2
 2260 ENDPROC
 2270 DEFPROCylwin
 2280 IF C%(M%(2))=C%(M%(3)) THEN PRO
Cmessage("TWO TO YELLOW",12,30):score
2%=score2%+2 ELSE PROCmessage("ONE TO
 YELLOW",12,30):score2%=score2%+1
 2290 A%=1
 2300 ENDPROC
 2310 DEFPROCchplayer
 2320 IF A%=2 THEN A%=1:ENDPROC
 2330 IF A%=1 THEN A%=2:ENDPROC
 2340 VDU4:PRINTTAB(3,30);SPC(34):VDU
5:RETURN
 2350 VDU4:PRINTTAB(3,1);SPC(34):VDU5
:RETURN
 2360 VDU18,3,3,240,241,10,8,8,18,3,A
%,242,243,18,3,3,10,8,8,244,245:RETUR
N
 2370 IF W%=1 OR H%=1 THEN VDU247:RET
URN ELSE VDU246:RETURN
 2380 DEFPROCfin
 2390 VDU4,22,7
 2400 VDU23;8202;0;0;0;
 2410 IF score1%>score2% PRINTTAB(7,1
2);"BLUE WINS  21 chalks to ";score2%
 ELSE PRINTTAB(7,12);"YELLOW WINS  21
 chalks to ";score1%
 2420 TIME=0:REPEAT UNTIL TIME>300
 2430 PRINTTAB(6,20)CHR$136;"PRESS AN
Y KEY TO PLAY AGAIN"
 2440 G=GET:RUN
 2450 DEFPROCslow
 2460 IF S%<0 S%=-1 ELSE S%=1
 2470 IF T%<0 T%=-1 ELSE T%=1
 2480 ENDPROC
 2490 REPORT:PRINTERR:PRINTERL
```

Lay your cards on the table



By STUART MENEFY

PROCEDURES

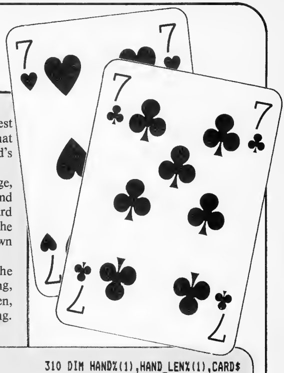
PROC_DEFINE	Defines characters, envelopes, sets up DIMs and major variables.
PROC_INIT	Sets up variables for each new game.
PROC_SCREEN	Displays screen.
PROC_GO	Decides who starts.
PROC_SHUFFLE	Sets up and shuffles the pack.
PROC_SOUND	Makes a bleep, when an input is not valid.
PROC_DELETE	Removes last character from input string.
PROC_COMMENT	Displays a centralised comment for required period of time.
PROC_HUMAN	Accepts and validates the humans entry, and performs the required function, if possible.
PROC_POSSIBLE	Finds out if a card may be played from the current hand.
PROC_DETAILS	Calculates the representation, value and suit of a card.
PROC_COMPUTER	Plays a card if possible from the micro's hand.
PROC_PLAY	Plays a card, and removes it from the hand.
PROC_FLIP	Attempts to make the sound of a card being put down!
PROC_DIS_HAND	Displays the human's hand.
PROC_ADVICE	Tells the human which card to play, using the micro's criteria.
PROC_SWAP	If possible, swaps the human's and micro's hands.
PROC_INSTRUCTIONS	Displays brief instructions, and asks for the human's name, the time for which messages are displayed, and if hands can be swapped.

FUNCTIONS

FN_DEAL	Moves a card from the top of the pack to one of the hands.
FN_INPUT	Asks the human for his/her instruction.
FN_COMPUTER	Returns the position of the 'best' card to play.
FN_END	Displays which player won, makes an appropriate sound, and asks if the human would like another go.
FN_YES_NO	Only returns TRUE if "Y" pressed, FALSE if "N" pressed.

VARIABLES

CARD$(13)	Contains the representation of each card.
C%(8)	The last card played in each section (see below).
CARD$	A representation of a card, including colour control characters, as determined in PROC_DETAILS.
CARD%	The cards face value, from 1 to 13, as determined in PROC_DETAILS.
DELAY%	Delay while messages are displayed, in centiseconds.
end%	Whether the game has finished or not.
G%(8)	In which section the most cards can be played, used to determine the micro's go.
GO%	The current player (see below).
GO_POS%	The position in the current hand of the best card to play
GO$	Command entered by human.
HAND%(1)	Position in memory where each hand starts.
HAND_LEN%(1)	Length of each hand.
NAME$	Human's name.
NUM_GO%	Number of cards that can be played.
PH%(8)	Position in the hand of each card that can be played.
PACK%	Position of the start of the pack in memory.
PACK_LEN%	Number of cards in the pack.
P1% & P2%	Used in shuffling the pack.
PH%	Position in the human's hand of the card he/she wants to play.
SCORE0% & SCORE1%	Number of games won by the human, and micro.
SNUM%	Number of times the human may swap hands with the micro.
SUIT%	Suit number, from 1 to 4, as determined in PROC_DETAILS.
VLU%	Card's internal value, as determined in PROC_DETAILS.



SEVENS is a traditional card game, depending partly upon luck and partly upon the skill of the players.

The game is played with a normal pack of playing cards, the object being to lay down all your cards before your opponent.

However cards may only be laid down in order, starting from seven, going up to king, and down to ace, and may only be laid down with other cards of the same suit.

Each player starts with seven cards, dealt from the pack, and puts down a card in turn. If he or she is unable to go, then a card must be picked up from the pack, and added to the hand.

In this way, four piles of cards develop, in order, and the winner is the player who is able to put down his or her final card.

In this version you play against the micro, who cannot "see" your hand, although it is displayed on the screen.

The player can also swap hands with the micro a limited number of times – as set by SNUM% – and can also give advice on which card to play.

The logic behind deciding which card to play is simple, as can be seen in FN_COMPUTER, and is used by the micro for its own goes, and also to give advice.

Having first checked to see how many cards can possibly be played (PROC_POSSIBLE), the micro returns a –1 if no card can be played. If only one card may be played then its position is returned, otherwise the remaining logic comes into play.

This first checks to see how many cards are held in the hand in each 'section'. There are eight sections, two for each suit, which extend from seven (which is included in both sections), either up or down.

The section that has the greatest number of cards in, and has a card that can be played, is used, and that card's position in the hand is returned.

This gives the player one advantage, in that the micro does not understand 'blocking'. This is when you hold a card for as long as possible, to prevent the micro being able to lay down his own cards, which come after it.

For example, if it is thought that the micro holds the jack, queen and king, then by delaying putting down the ten, you stand a greater chance of winning.

In the arrays and scoring, the human is player 0, and the micro is player 1.

Also, the word "section" in arrays C%(8) and G%(8), section 1 (or element 1) is from seven hearts, to the king hearts, 2 is from the ace hearts to the seven hearts, and similarly 3 and 4 are clubs, 5 and 6 diamonds, and 7 and 8 spades.

```
   10 REM    SEVENS
   20 REM by S.I.Menefy
   30 REM  for 32K BBC
   40 REM (c) Micro User
   50 MODE7
   60 PROC_DEFINE
   70 PROC_INSTRUCTIONS
   80 PROC_INIT
   90 MODE 1
  100 PROC_SCREEN
  110 PROC_GO
  120 PROC_SHUFFLE
  130 REPEAT
  140 IF GO% PROC_COMPUTER ELSE PROC_
HUMAN
  150 UNTIL HAND_LEN%(0)=-1 OR HAND_L
EN%(1)=-1 OR end%
  160 IF NOT(end%) THEN IF FN_END THE
N 80
  170 MODE 7
  180 *FX4
  190 END
  200 DEF PROC_DEFINE
  210 VDU 23,224,54,127,127,127,62,28
,8,0
  220 VDU 23,225,8,28,62,127,127,28,6
2,0
  230 VDU 23,226,8,28,62,127,62,28,8,
0
  240 VDU 23,227,28,28,107,127,107,8,
28,0
  250 VDU 23,228,0,0,0,0,0,96,96,0
  260 ENVELOPE 1,1,2,0,0,60,0,0,127,0
,0,-127,126,126
  270 ENVELOPE 2,3,0,0,0,0,0,0,121,-1
3,0,-10,120,120
  280 ENVELOPE 3,0,-2,0,0,100,0,0,1,0
,0,-1,1,0
  290 ON ERROR REPORT:PRINT" at line 
";ERL:END
  300 SCORE0%=0:SCORE1%=0
  310 DIM HAND%(1),HAND_LEN%(1),CARD$
(13),C%(8),G%(8),PH%(8):PACK%=&900
  320 HAND%(0)=PACK%+52:HAND%(1)=PACK
%+104
  330 FOR A%=1 TO 13:READ CARD$(A%):N
EXT A%
  340 ENDPROC
  350 DATA ACE,2,3,4,5,6,7,8,9,10,JAC
K,QUEEN,KING
  360 DEF PROC_INIT
  370 ON ERROR IF ERR=17 THEN 130 ELS
E REPORT:PRINT" at line ";ERL:END
  380 end%=FALSE
  390 SNUM%=2

  400 PACK_LEN%=51:HAND_LEN%(0)=-1:HA
ND_LEN%(1)=-1
  410 ENDPROC
  420 DEF PROC_SCREEN
  430 VDU 19,2,2;0;
  440 VDU 23,0,10,32,0;0;0;24,0;336;9
76;944;18,0,130,16,26
  450 PRINT"Games to "NAME$" ";SCORE0
%
  460 PRINT"Games to computer ";SCORE
1%
  470 RESTORE 520
  480 FORA%=1 TO 4
  490 READ Y%
  500 MOVE 0,Y%:DRAW 976,Y%
  510 NEXT
  520 DATA 880,432,336,208
  530 MOVE 0,944:DRAW 1280,944
  540 MOVE 976,1024:DRAW 976,0

  550 FOR A%=0 TO 3:MOVE A%*224+80,94
4:DRAW A%*224+80,432:NEXT A%
  560 VDU 5
  570 FOR A%=1 TO 13
  580 MOVE 16+(16*(LEN(CARD$(A%))=2))
```

→

From Page 55

```
,A%*32+440
  590 PRINT LEFT$(CARO$(A%),1-(LEN(CA
RD$(A%))=2))
  600 NEXT A%
  610 FDR A%=1 TO 4
  620 MOVE A%*224-48,928
  630 GCOL0,A% MDO 2
  640 PRINT CHR$(A%+223)
  650 NEXT A%
  660 GCOL 0,3
  670 MOVE 984,1000
  680 PRINT"Your hand"
  690 VDU 4
  700 PRINT TAB(0,22)"Number left in
pack 52"
  710 PRINT"Number in computers hand
7"
  720 IF S% PRINT"Number of swaps lef
t 2" ELSE PRINT"You cannot swap hands
"
  730 PRINT'TAB(13)"KEYS"
  740 PRINTTAB(2,27)"Cannot go"
  750 PRINT"  End game"
  760 PRINT"  Advice on what to do"
  770 IF S% PRINT"  Swap hands with t
he computer"
  760 PRINT"or number of the required
 card";
  790 COLOUR 1
  800 PRINTTAB(0,27)"C"'"E"'"A"'CHR$(
-63*S%)
  610 CDLOUR 3
  820 CDLOUR 130
  630 ENOPROC
  840 DEF PROC_GO
  850 IF SCORE0%=SCORE1% GO%=RNO(2)-1
 ELSE IF SCORE0%<SCDRE1% GO%=0 ELSE G
O%=1
  860 IF GD% PRINT TAB(10,19)"I go fi
rst" ELSE PRINT TAB(8,19)"You go firs
t"
  870 ENOPROC
  880 OEF PROC_SHUFFLE
  890 SDUNO 0,-13,3,60
  900 SDUNO 1,3,255,60
  910 A%=1
  920 FOR B%=0 TO 51
  930 IF A% MOD 20=14 A%=A%+7
  940 PACK%?B%=A%
  950 A%=A%+1
  960 NEXT B%
  970 FOR A%=1 TO 50+RNO(50)
  980 P1%=RNO(52)+PACK%-1:P2%=RND(52)
+PACK%-1
  990 T%=?P1%
 1000 ?P1%=?P2%
 1010 ?P2%=T%
 1020 NEXT A%
 1030 FOR A%=0 TO 6
 1040 FOR B%=0 TO 1
 1050 C%=FN_OEAL(B%)
 1060 NEXT B%
 1070 NEXT A%
 1080 PRINT TAB(0,20)SPC(30)
 1090 PROC_OIS_HAND
 1100 ENDPROC
 1110 OEF FN_DEAL(C%)
 1120 IF PACK_LEN%=-1 =FALSE
 1130 L%=-1
 1140 REPEAT
 1150 L%=L%+1
 1160 UNTIL HANO%(C%)?L%>PACK%?PACK_L
EN% OR L%>HANO_LEN%(C%)
 1170 FOR K%=HANO%(C%)+HANO_LEN%(C%)
TO HANO%(C%)+L% STEP-1:?(K%+1)=?K%:NE
XT
 1180 HANO_LEN%(C%)=HANO_LEN%(C%)+1
 1190 HANO%(C%)?L%=PACK%?PACK_LEN%
 1200 PACK_LEN%=PACK_LEN%-1
 1210 COLOUR 128
 1220 PRINT TAB(20,22);PACK_LEN%+1;"
"
 1230 COLDUR 130
 1240 PROC_FLIP
 1250 =TRUE
 1260 OEF FN_INPUT
 1270 *FX15
 1280 COLOUR 3
 1290 VDU 23,0,10,96,0;0;0;
 1300 INPUTTAB(0,19)"Please enter you
r go"SPC(10)TAB(21,19)input$
 1310 VDU 23,0,10,32,0;0;0;
 1320 IF VAL(input$)>HANO_LEN%(0)+1 P
ROC_SOUNO:PRDC_COMMENT("You don't hav
e "+input$+" cards",FALSE):BOTO 1270
 1330 IF input$="" PROC_SOUND:GOTO 12
70
 1340 =input$
 1350 OEF PROC_SOUND
 1360 SDUNO 1,-15,101,2
 1370 ENDPROC
 1380 OEF PROC_DELETE
 1390 IF input$=""ENDPROC
 1400 input$=LEFT$(input$,LEN(input$)
-1)
 1410 VOU 127
 1420 ENDPROC
 1430 DEF PROC_CDMMENT(COMMENT$,COL%)
 1440 PRINT TAB((30-(LEN(COMMENT$)+CO
L%*5))/2,20)COMMENT$;
 1450 A%=INKEY(DELAY%)
 1460 PRINT TAB(0,20)SPC(30)
 1470 ENDPROC
 1480 DEF PROC_HUMAN
 1490 PROC_POSSIBLE
 1500 GO$=FN_INPUT
 1510 IF LEFT$(GO$,1)<>"C" THEN 1590
 1520 IF NUM_GO%>0 PROC_COMMENT("You
can go",FALSE):GOTD 1500
 1530 G%=FN_OEAL(0)
 1540 PRDC_OETAILS(L%,0)
 1550 IF G% PROC_COMMENT("You picked
up "+CARO$,TRUE) ELSE PROC_COMMENT("T
he pack is empty",FALSE)
 1560 PROC_OIS_HANO
 1570 BO%=1
 1580 ENOPROC
 1590 IF LEFT$(GO$,1)="A" ANO NUM_GO%
>0 PROC_ADVICE:GDTO 1500
 1600 IF LEFT$(GO$,1)="A" THEN 1650
 1610 IF LEFT$(GO$,1)="S" PROC_SWAP:G
OTO 1490
 1620 IF LEFT$(GO$,1)="E" end%=TRUE:E
NDPROC
 1630 PH%=VAL(GD$)-1
 1640 IF PH%=-1 PROC_SOUNO:PRDC_CDMME
NT("Pardon?",FALSE):GOTO1500
 1650 IF NUM_GO%=0 PRDC_COMMENT("You
can't go",FALSE):GDTD 1530
 1660 PROC_OETAILS(PH%,0)
 1670 FOR A%=1 TO NUM_GO%
 1680 IF PH%=PH%(A%) THEN A%=9
 1690 NEXT A%
 1700 IF A%<9 PROC_COMMENT("You canno
t play that card",FALSE):GDTO 1500
 1710 PROC_PLAY(PH%)
 1720 PROC_DIS_HAND
 1730 GO%=1
 1740 ENDPROC
 1750 DEF PROC_POSSIBLE
 1760 NUM_GO%=0
 1770 FOR A%=0 TD HAND_LEN%(GO%)
 1780 PROC_OETAILS(A%,GO%)
 1790 IF C%((SUIT%*2)+(CARO%>6))=VLU%
-1 AND CARD%>6 THEN NUM_GO%=NUM_GD%+1
:PH%(NUM_GO%)=A%
 1800 IF C%((SUIT%*2)+(CARO%>6))=VLU%
+1 AND CARO%<=6 THEN NUM_GD%=NUM_BO%+
1:PH%(NUM_GO%)=A%
 1610 IF CARO%=7 THEN NUM_GO%=NUM_GO%
+1:PH%(NUM_GO%)=A%
 1620 NEXT A%
 1830 ENOPROC
 1840 OEF PROC_OETAILS(pos%,D%)
 1850 VLU%=HANO%(D%)?pos%
 1860 SUIT%=(VLU% DIV 20)+1
 1870 CARD%=VLU% MOD 20
 1880 CARD$=CHR$ 17+CHR$(SUIT% MOD 2)
+CHR$ 18+CHR$ 0+CHR$(SUIT% MOD 2)+CAR
D$(CARO%)+CHR$(223+SUIT%)
 1890 ENOPROC
 1900 DEF FN_COMPUTER
 1910 PRDC_PDSSIBLE
 1920 IF NUM_GD%=1 =PH%(1)
 1930 IF NUM_GD%=0 =-1
 1940 FOR A%=1 TO 8:G%(A%)=0:NEXT A%
```

→

mixed moods
two amazing new games from Software Invasion . . .

Software Invasion games are available from WH SMITH, HMV, all SPECTRUM SHOPS and over 1,000 retail outlets.

3D Grand Prix

This is the game you've been waiting for! A 3D high speed formula 1 car race round 8 different international circuits.

Just look at these amazing features:–
* Bends arranged to simulate 8 actual Grand Prix circuits.
* Working cockpit – Rev. counter, Speedometer, Temperature Gauge, gears, moving steering wheel.
* Other cars which alter in perspective and size as they overtake. * Signposts. * Realistic sound effects such as engine noise, skids and crunching shunts. * Keyboard or Joystick control. * Volume control.

Runs on O.S. 1.0 or greater. More than a year of man hours have gone into creating this superb game.

BBC MICRO MODEL B
£9.95 CASSETTE
£11.95 DISK
Authors: Mick O'Neill and Dave Mendes

Blitzkrieg

NOW WITH COLOUR GRAPHICS! (BBC VERSION ONLY)

Blitzkrieg is a 3D Tank Battle from the same author as our Best Seller "3D Bomb Alley".

This program is probably unique in the fact that all the graphics were created with the help of models, a video camera, a video digitiser and some very special Light Pen software.

This has resulted in extremely fine detail and realism, with fast moving real time graphics – you don't have to wait for the screen to be updated between each move.

Features include:–
* Detailed scrolling landscape. * Rotating tanks.
* Detailed explosions. * 3D shell movement. * Recoiling barrel. * 3D sound. * Keyboard/Joystick option.
* Freeze game facility.
* Optional sound.
* Demo' mode.
* High score & ranking system.

BBC MICRO MODEL B
£7.95 CASSETTE
£9.95 DISK
Author: Simon Vout.

SOFTWARE INVASION

EMS

50 Elborough Street, Southfields, London SW18 5DN

HOW TO ORDER

You may purchase any of the games above direct. All you have to do is write your name and address on a piece of paper, item(s) required, enclosing your cheque/P.O. made payable to SOFTWARE INVASION.
Please allow 7 to 14 days for delivery (orders are normally dispatched within 48 hours!)
OVERSEAS ORDERS. Please add £1.00 per game

DISTRIBUTORS, MULTIPLE STORES, RETAILER GROUP AND EXPORT INQUIRIES INVITED.

Do you write your own programs? If you have an unusual program which can meet our standards you could be earning a substantial sum every week! Why not take advantage of our sought after reputation. Write now!

other SOFTWARE INVASION GAMES . . .send for full list now!

The company which brought you the first self-build Arcade game and the first Adventure with sound, just had to be the one to give you the very first QUEST for the Beeb!

Swords and Sorcery

a new experience...

QUEST programs originated on the American mainframe computers, and were converted to micros, though requiring so much memory could only be made to work on expanded Apple micros and the like. Now, Kansas have crammed a full feature Quest into 32K, giving every BBC Micro owner the opportunity to play and experience these unique games.

So what is a Quest? Think of an Adventure, then think of very much more. A Quest is operated similar to an Adventure, but no longer are you alone, starting with three helpers. As you progress you will encounter many other characters, but unlike an Adventure program were all characters are passive, these can be either hostile, friendly or indifferent. If hostile you have to fight, and quickly; if friendly they may join your team bringing more strength, magical ability and carrying capability; if indifferent you could perhaps try a bribe (but beware if you offer too little) or you could sell an object and raise yourself more cash as well as lessening your load.

But most of the effort is your own, with treasure, money, magic and strength all having a bearing on your progress. At times it will pay to be vicious and abandon your friends as they become weaker, or even go in fighting before characters show their true colours, the element of surprise using less strength.

As can be expected in Swords and Sorcery there is a story:

Once upon a time in a far away land called Iriuma a magical Princess cast a spell of banishment on the wicked Sorcerer Brogfelt. However, just before the spell took effect, Brogfelt threw his arms into the shape of changing and cast a return spell, changing princess Illear into a diamond. Taking the diamond with him, Brogfelt took sanctuary in the Dungeons of Terror amongst the various monsters and demi-humans. Here he split the diamond into four parts and changed each part into a different crystal. Brogfelt then hid each crystal in the dungeons never to be found again. When the king heard of this he summoned all his faithful Knights to him and offered half his lands for the four crystals. Sadly they all perished in the Dungeons of Terror attempting it. Now the king has offered anyone his other daughter's hand together with half his lands. Hearing this, you set off to the king's palace to offer your services. The king is astounded but nevertheless offers you a party of three prisoners from his jail, promising them a full pardon if they will go and aid you. As you leave the king stops you and thrusts a scrap of paper into your hand explaining that it contains the location of each part of the crystal from the entrance of the dungeons. It was written very shakily and stained in blood. The king says: "Let me introduce you to the three prisoners that I have volunteered to go along with you..."

The characters you meet include a Troll, Orc, Thief, Dwarf, Goblin, Madman, Witch, Hobgoblin, Mad Monk and of course the wicked wizard himself, all in fact you would expect in a magical Swords and Sorcery...

All the objects have a use, but be careful, for picking the Dragons Tooth could be fatal; though if you find the Staff of Healing try and get it; an Idol of a forgotten God should be left well alone; the Ring may help you; the Magic Axe certainly will; the Old Book will give some clues; rub the Glass Ball; used properly the Magic Carpet will get you out of trouble; the Rolled Scroll too is useful; but not so the Fools Gold; be careful with the Bottle of Liquid; but drink the Magic Potion; treat the Golden Orb with care; but of course the Sword is the greatest help of all. And so it goes on, and on and on...

Unlike an Adventure game, were once you have solved the plot, and it is all finished for good, Swords and Sorcery generates an entirely different scenerio every time, even though totally logical! But if you really intend to see a game right through to the end, the game saving facility allows you to do this, playing the same scenerio, time after time, to its end. It is so different from an Adventure, that it actually has nine–yes nine– levels of play, with the ninth having so many locations and of such complexity, that we would be amazed if anybody ever solved it!

If you are an Adventure addict, this will really spoil you! If you do not care for Adventures the activity in this unique game will suit you as well. It is one of those you just will not be able to leave alone...

Though there are many Adventure games using the word Quest in their title, these are not Quests in this, the true sense of the word, but are just ordinary Adventures.

FEATURES a Map routine showing exactly where you are and where you have explored.

FEATURES colour denoting the different aspects of the game.

FEATURES sound on or sound off facility.

FEATURES entirely different scenerio for every game and all totally logical

FEATURES nine levels of difficuly from fairly easy to impossible.

FEATURES game saving routine to be able to replay the same scenerio.

Only available from Kansas at £10.35 Vat and post paid.

Our normal return first class post service applies, with same day postage if ordered on Access or Barclaycard.

This program carries our famed lifetime guarantee.

Available on cassette only for BBC model B.

Kansas City Systems, Unit 3, Sutton Springs Wood, Chesterfield, S44 5XF.
Tel. 0246 850357

From Page 56

```
 1950 FOR A%=0 TO HAND_LEN%(GO%)
 1960 PROC_DETAILS(A%,GO%)
 1970 G%((SUIT%*2)+(CARD%>6))=G%((SUI
T%*2)+(CARD%>6))+1
 1980 NEXT A%
 1990 MAX%=0
 2000 FOR A%=1 TO NUM_GO%
 2010 IF G%(A%)>MAX% MAX%=G%(A%)
 2020 NEXT A%
 2030 FOR A%=1 TO NUM_GO%
 2040 PH%=PH%(A%):PROC_DETAILS(PH%,GO
%)
 2050 IF (SUIT%*2)+(CARD%>6)=MAX% OR
(CARD%=7 AND (SUIT%*2)=MAX%)A%=9
 2060 NEXT A%
 2070 =PH%
 2080 DEF PROC_COMPUTER
 2090 PRINT TAB(0,19)SPC(30)
 2100 GO_POS%=FN_COMPUTER
 2110 COLOUR 3
 2120 IF GO_POS%=-1 PRINT TAB(8,19)"I
 could not go";:G%=FN_DEAL(1):GOTO 21
60
 2130 PROC_DETAILS(GO_POS%,1)
 2140 PRINT TAB(13-(LEN(CARD$)/2),19)
"I played ";CARD$;
 2150 PROC_PLAY(GO_POS%)
 2160 COLOUR 128:COLOUR 3
 2170 PRINT TAB(25,23);HAND_LEN%(1)+1
;" "
 2180 COLOUR 130
 2190 A%=INKEY(DELAY%)
 2200 GO%=0
 2210 ENDPROC
 2220 DEF FN_END
 2230 COLOUR 3
 2240 IF HAND_LEN%(0)=-1 PRINT TAB(31
,3)SPC(9);TAB(0,19)"  Congratulation
s, you win":SCORE0%=SCORE0%+1:SOUND 1
,1,100,35:GOTO 2350
 2250 PRINT TAB(0,19)SPC(8)"Bad luck
I won"SPC(8):SCORE1%=SCORE1%+1
 2260 FOR B%=8 TO 0 STEP -8
 2270 RESTORE 2420
 2280 FOR A%=1 TO 5
 2290 READ N%,D%
 2300 IF A%=5 AND B%=0 D%=30
 2310 SOUND &101,2,N%+B%,D%
 2320 SOUND &102,2,N%+B%+48,D%
 2330 SOUND 1,0,0,2
 2340 NEXT A%:NEXT B%
 2350 COLOUR 128
 2360 PRINT TAB(10+LEN(NAME$),0);SCOR
E0%
 2370 PRINT TAB(18,1);SCORE1%
 2380 FOR A%=1 TO 8:C%(A%)=0:NEXT A%
 2390 COLOUR 130
 2400 PRINT TAB(0,20)"Do you want to
play again(Y/N)"
 2410 =FN_YES_NO
 2420 DATA 4,10,4,10,12,15,4,5,0,20
 2430 DEF PROC_PLAY(PLAY_POS%)
 2440 PROC_DETAILS(PLAY_POS%,GO%)
 2450 VDU 5
 2460 MOVE ((SUIT%-1)*224)+((256-(LEN
(CARD$)*32)/2))+16,440+(CARD%*32):PRI
NTCARD$
 2470 VDU 4
 2480 C%((SUIT%*2)+(CARD%>6))=VLU%
 2490 IF CARD%=7 C%(SUIT%*2)=VLU%
 2500 FOR A%=HAND%(GO%)+PLAY_POS% TO
HAND%(GO%)+HAND_LEN%(GO%)-1
 2510 ?A%=A%?1
 2520 NEXT
 2530 HAND_LEN%(GO%)=HAND_LEN%(GO%)-1
 2540 PROC_FLIP
 2550 ENDPROC
 2560 DEF PROC_FLIP
 2570 REPEAT UNTIL ADVAL(-5)=15
 2580 SOUND 0,-12,5,0
 2590 SOUND 0,-13,4,0
 2600 SOUND 0,0,0,6
 2610 ENDPROC
 2620 DEF PROC_DIS_HAND
 2630 VDU 24,980;0;1279;940;:CLG
 2640 FOR A%=0 TO HAND_LEN%(0)
 2650 PROC_DETAILS(A%,0)
 2660 COLOUR 3
 2670 PRINT TAB(31-(A%<9),A%+3);A%+1;
CHR$(228);CARD$
 2680 NEXT A%
 2690 VDU 26
 2700 ENDPROC
 2710 DEF FN_YES_NO
 2720 REPEAT
 2730 A$=GET$
 2740 UNTIL A$="Y" OR A$="N"
 2750 =A$="Y"
 2760 DEF PROC_ADVICE
 2770 GO_POS%=FN_COMPUTER
 2780 PROC_DETAILS(GO_POS%,0)
 2790 PROC_COMMENT("I advise you to p
lay "+CARD$,TRUE)
 2800 ENDPROC
 2810 DEF PROC_SWAP
 2820 IF S%=0 OR SNUM%=0 PROC_COMMENT
("You cannot swap hands.",FALSE):ENDP
ROC
 2830 T%=HAND%(0):HAND%(0)=HAND%(1):H
AND%(1)=T%
 2840 T%=HAND_LEN%(0):HAND_LEN%(0)=HA
ND_LEN%(1):HAND_LEN%(1)=T%
 2850 COLOUR 128
 2860 PRINT TAB(25,23);HAND_LEN%(1)+1
;" "
 2870 COLOUR 130
 2880 PROC_DIS_HAND
 2890 SNUM%=SNUM%-1
 2900 COLOUR 128
 2910 COLOUR 3
 2920 PRINTTAB(21,24);SNUM%
 2930 COLOUR 130
 2940 ENDPROC
 2950 DEF PROC_INSTRUCTIONS
 2960 FOR A%=0 TO 1
 2970 VDU 31,0,A%,134,157,132,141:PRI
NTSPC(10)"Sevens"
 2980 NEXT A%
 2990 PRINT"   The object of the game
 is to put all"
 3000 PRINT" the cards in your hand d
own before the"
 3010 PRINT" computer can. You may on
ly lay one card";
 3020 PRINT" down at a time, starting
 from seven and";
 3030 PRINT" going up and down to Kin
g and Ace."
 3040 PRINT'"   Your cards are displa
yed next to a"
 3050 PRINT" number, and it is this n
umber that you"
 3060 PRINT" enter to play the card.
If you cannot"
 3070 PRINT" go type ""C"" and you wi
ll be dealt a"
 3080 PRINT" card."
 3090 PRINT"   If you want advice on
which card to"
 3100 PRINT" play type ""A"" and you
will be told."
 3110 PRINT"   If you wish, you can c
hose to swap"
 3120 PRINT" your hand with the compu
ter, but only"
 3130 PRINT" two times during the gam
e."
 3140 *FX15
 3150 *FX4 1
 3160 VDU 23;10,96,0;0;0;
 3170 PRINTTAB(1,20)"Please type in y
our name ((<19 letters)"
 3180 INPUT TAB(1,21)SPC(40)TAB(1,21)
NAME$
 3190 NAME$=LEFT$(NAME$,18)
 3200 PRINTTAB(1,20)"Please type how
long messages are to be";
 3210 PRINT" displayed for (in second
s) "
 3220 INPUTTAB(28,21)SPC(12)TAB(28,21
)A$
 3230 DELAY%=VAL(A$)*100
 3240 PRINTTAB(1,20)"Do want to be ab
le to swap your hand   ";
 3250 PRINT" with the computer. (Y/N)
";SPC(15);TAB(26,21);
 3260 S%=FN_YES_NO
 3270 ENDPROC
```

Play Cribbage

Peg your way round ALAN FARMAN's version of the classic card game

THIS version of cribbage pits one player against the BBC Micro. For those unfamiliar with crib, the object of the game is to "peg" your way round a board by scoring points. There are two main ways to score.

Firstly you can score points as you and your opponent play your cards alternately – this is the play.

Secondly, after the play, you lay down your hand and gain points according to the various scoring combinations you hold – this is the lay.

All cards are worth their face value with jack, queen and king counting ten. Ace is low.

Play alternates between player and computer, with the dealer of the first game being picked at random. In practice the micro deals the card.

However the deal is considered to alternate between micro and player, the non-dealer having to lay the first card in a game.

The player is dealt six cards, from which he has to retain four for his hand. The two cards he discards are thrown into the "box" or "crib". These, together with the computer's discards, form an extra hand.

This hand is scored at the lay and the points awarded to the dealer – that is, this extra hand alternates between player and micro.

Thus when selecting discards you have to bear in mind whose box it is. You must also try to ensure that the hand you are left with is balanced enough to provide points in both play and lay.

Once the discards have been made a starter card is dealt from the pack. This is the "turn up". If it's a jack, the dealer scores two points. This turn up plays no further part in the play, but is considered to be an integral part of the hand during the lay.

The non-dealer then plays a card followed by the dealer, the total points value of the cards being added. Play continues alternately until the total is either exactly 31, or neither player can play a card without exceeding that value – or until all the cards in the hand have been played.

If, after your turn, your opponent cannot go, you are still entitled to if you can do so legally, scoring as normal.

If a position is reached where neither player can go, that round of play is finished and a new one started with the remaining cards, the player who first failed to play a card going first.

The last player to play a card in a round scores points: two if he reaches 31 exactly, otherwise one. However there are also other points to be won during the play.

You score two points if you bring the total to 15, or if your card forms a pair with the previous card. If your card forms three the same you score six, and all four identical scores 12.

Also, if the card you play forms a run of three or more (including your opponent's cards), you score one for each card in the run. Note that they don't have to be in order. If you play a four then the micro plays a six and you then play a five, you score three for the run, even though it's 4, 6, 5.

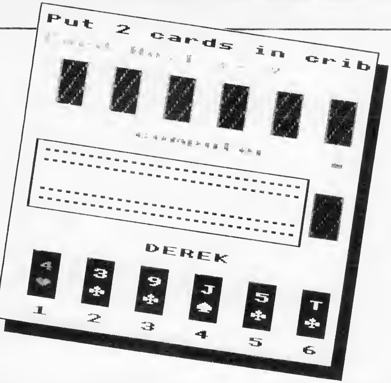
Incidentally, if your four had been the first card in the round, you would score five, since the total would be 15 – you score two for that and three for the run. All you have to hope is that the micro doesn't have a three – then again you might have a two to follow it . . .

When all the cards have been played they are laid out and the cards in each hand – together with the turn up – are scored.

You score two for every combination of 15 you can find in your hand, and

PROCEDURES

PROCINIT	Rules.
PROCTAPE	Chains game.
PROCget	Pause control.
PROCT	Time delay.
PROC_vdu_chars	Sets up characters.
PROCplayname.	Player's name.
PROCBOARD	Draws board.
PROCDEALER	Randomly decides first dealer.
PROC_varis	Sets up variables.
PROCSHUFFLE	Shuffles cards.
PROCCARDS	Draws cards on screen.
PROCALLOCATE	Allocates CARD$ type to numeric. CARD% for calculation purposes.
PROCSTARTER	Draws starter or crib card.
PROCSTARTERBACK	Draws starter card back.
PROCBOXCARD	Allocates the four cards to the crib.
PROCWHOLAY	Sorts out who is to lay first card.
PROCGAME	Plays the game.
PROCCARDFRONT	Draws front of card.
PROCCARDBACK	Draws back of card.
PROCUP & PROCDOWN	Moves box character about screen.
PROCNUMBER	Converts random number into card number used when sorting out dealer.
PROCsound	Makes a sound.
PROCPACK	Reads 52 cards into an array and allocates suit.
PROCPLAYERCARD	Draws player's cards on screen.
PROCcardcol	Sorts out player's suit colour.
PROCCPBEST & PROCCPSECBT	Sorts out computer's best or second best cards according to whether Level 1 or 2 is being played.
PROCcompsc & PROCplaysc	Draws scores on screen for both computer and player as play proceeds.
PROCGAMETOT	Displays game score.



similarly for every pair. Three of a kind count six, and four count 12. Runs of three or more count one for each card involved.

If all four cards in your hand are the same suit you score four for the flush. (In the box, you need all four cards in the hand plus the turn up to be of the same suit for a flush – and you score five for it.)

Also, if you have a jack in your hand of the same suit as the turn up, you score "one for his nob".

The non-dealer scores his lay down first, followed by the dealer, who then also totals his box. This is quite important, as it lets the non-dealer have first chance of pegging out – that is, pegging his way around the scoring board twice.

There are two levels of play. At level one, the computer has limited search facilities and this level should be used to learn the game.

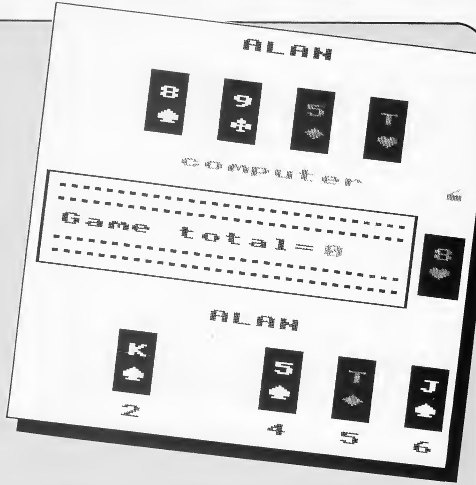
At level two a full search of any four combinations from six cards is carried out, making it suitable for the more experienced player.

PROCCOMPCHECK	Part of computer's search routine to find best cards.
PROC(S)–FIVECARD, FOURCARD, THREECARD, TWOCARD, SAME, PAIR, JACK	These procedures search for runs, 15s, flushes, pairs etc.
PROCSWOP	Swops computer/player cards to enable search for runs.
PROCKEEP	Decides which four cards computer should keep.
PROCCOMPGAMECARD	Computer plays with these cards.
PROCCOMPFIRSTMOVE	Plays computer's first move (only if computer goes first).
PROCPLAYERMOVE	Plays player's moves.
PROCCOMPNEXTMOVE	Plays computer's subsequent moves.
PROCODSANSDS	Clears up a bit of logic.
PROCPLAYCARDGONE/ COMPCARDGONE	Removes cards from game screen.
PROCFACECARD	Prints card face.
PROCPLAYFIRST/ COMPFIRST	Decides who scores first, displays cards again and scores.
PROCCLEARUP	Lays a card and sorts out specific variables.
PROCLAY	Lays a card in CLEARUP above.
PROCPLAYCARDHAND	Re-allocates player's cards before game commences; enables player's scoring to be carried out by the computer.
PROCcorner	Prints arrow character when scoreboard turns corner (useful on B/W sets).
PROCWINNER	Declares winner.
PROCBOX	Displays boxcards.
PROCPLCOSCCHECK	Part of score checking routine.
PROCCOMPBOX	Sorts out boxcards.
PROCscore	Shows total score held by each player at the end of each hand.
PROCblank_score	Removes total score from game board.

```
   10 REM *** CRIBBAGE ***
   20 REM *  Alan Farman *
   30 REM *(c) Micro User*
   40 REM ***   1984   ***
   50 ON ERROR GOTO 2380
   60 DIM CD$(18),BOX$(5),CD%(18),CHC
D%(18),KP%(3),X1%(4),X2%(4),Gone%(6),
run%(9):MODE7:PROCVD:PROCMN:seed%=(RN
D(-TIME)/LEN(NAME$))
   70 MODE5:PROCBD:PROCDL:CXM%=976:PX
M%=976:CYM%=600:PYM%=472:REPEAT:PROCV
A:PROCSH:PROCCD(1):PROCAL(1,7):PROCST
(" "):PROCSB:PROCBO:PROCWH:PROCGM:UNT
IL FALSE
   80 DEF PROCVD:VDU23,226,16,32,126,
33,17,1,1,126,23,227,0,0,60,36,36,60,
0,0,23,228,0,126,66,66,66,66,126,0,23
,229,255,129,129,129,129,129,129,255,
23,230,2,4,8,16,62,62,62,62,23,231,54
,127,127,127,62,28,8,0
   90 VDU23,232,8,28,62,127,127,127,2
8,62,23,233,8,28,62,127,62,28,8,0,23,
234,8,28,28,107,127,107,8,28:HEART$=C
HR$231:SPADE$=CHR$232:DIAMOND$=CHR$23
3:CLUB$=CHR$234:ST$=STRING$(59," "):S
T1$=STRING$(19," "):ENDPROC
  100 DEF PROCVA:nexcd%=0:plcdpo%=1:c
pnogo%=0:plnogo%=0:fiftn%=0:pair%=0:j
ack%=0:co%=0:pl%=0:npair%=0:newno%=31
:n%=0:point%=0:cpfin%=0:plfin%=0:plco
nt%=0:cpprecd%=0:plprecd%=0:total%=0:
corp%=1:ftpr%=0:twoetc%=0:thiscd%=0:l
aid%=0
  110 gotrun%=0:plcorun%=0:apair%=0:c
opr%=0:plpr%=0:FORI%=1TO6:Gone%(I%)=0
:NEXT:VDU28,0,3,19,0,23;11,0;0;0;0:EN
DPROC
  120 DEF PROCCF(X%,Y%):VDU25,4,X%;Y%
;25,5,X%;(Y%+150);25,5,(X%+100);(Y%+1
50);25,85,X%;Y%;25,5,(X%+100);Y%;25,8
5,(X%+100);(Y%+150);:ENDPROC
```

From Page 61

```
  130 DEF PRDC8D:VDU5,18,0,3:FORY%=44
0TD600STEP32:FDRX%=48TD976STEP32:IFY%
<=472 DRY%>=568THENMOVEX%,Y%:PRINT"."
  140 NEXT:NEXT:VDU18,0,3,25,4,32;384
;25,5,32;608;25,5,1040;608;25,5,1040;
384;25,5,32;384;25,4,3B4;688;1B,0,1:P
R1NT"computer":VDU18,0,2:MDVE 640-((L
EN(NAME$)/2)*64),320:PRINT NAME$:VDU1
8,0,3,4:ENDPRDC
  150 DEF PRDCST(n$):VDU5,18,0,3,25,4
,1079;421;25,5,1079;571;25,5,1179;571
;25,85,1079;421;25,5,1179;421;25,85,1
179;571;25,4,1100;504;18,0,0:PRINT n$
:VDU1B,0,3,4:ENDPRDC
  160 DEF PRDCCK(X%,Y%):VDU5,18,0,1,2
5,4,X%;Y%;25,5,(X%+100);(Y%+150);25,5
,X%;(Y%+150);25,5,X%;Y%;25,5,(X%+100)
;Y%;25,5,(X%+100);(Y%+150);25,4,(X%+5
0);(Y%+150);25,5,X%;(Y%+75);25,4,(X%+
100);(Y%+75);25,5,(X%+50);Y%;18,0,3,4
:ENDPROC
  170 DEF PRDCDL:PRDCST(" "):PRDCSB:C
OLDUR3:PRINT TAB(0,1)"First Jack for
crib":VDU7:PRDCT(1.5):REPEAT:PROCUP:P
RDCNU:PRDCST(N$)
  180 1FN$="J"THENPRDCT(.5):deal%=1:D
EALER$="computer":8DTD210ELSEPROCT(.5
)
  190 PRDCDN:PROCNU:PROCST(N$)
  200 1FN$="J"THENPROCT(.5):deal%=-1:
DEALER$=NAME$:GDTD210ELSEPRDCT(.5)
  210 UNTILN$="J":PROCsd:PRDCST(" "):
PRDCSB:PRINTTAB(0,0);ST$:CDLDUR1:PRIN
TTA8(10-(LEN(DEALER$)/2),0);DEALER$:C
DLDUR2:PRINTTAB(0,2);"has the first c
rib!";:PRDCT(4):ENDPROC
  220 DEF PRDCSB:PRDCCK(1079,421):END
PRDC
  230 DEF PRDCNU:J%=RND(13):1FJ%=1THE
NN$="A"
  240 IFJ%>1ANDJ%<10THENN$=STR$(J%)
  250 IFJ%=10THENN$="T"
  260 IFJ%=11THENN$="J"
  270 1FJ%=12THENN$="Q"
  2B0 1FJ%=13THENN$="K"
  290 ENDPROC
  300 DEF PRDCAL(a%,b%):FORN%=a%TOb%:
IFRIGHT$(CD$(N%),1)="A"THENCD%(N%)=1:
CHCD%(N%)=1
  310 IFRIGHT$(CD$(N%),1)="T"THENCD%(
N%)=10:CHCD%(N%)=10
  320 IFRIGHT$(CD$(N%),1)="J"THENCD%(
N%)=10:CHCD%(N%)=11
  330 IFRIGHT$(CD$(N%),1)="Q"THENCD%(
N%)=10:CHCD%(N%)=12
  340 IFRIGHT$(CD$(N%),1)="K"THENCD%(
N%)=10:CHCD%(N%)=13
  350 IFVAL(R1GHT$(CD$(N%),1))>1ANDVA
L(RIGHT$(CD$(N%),1))<10THENCD%(N%)=VA
L(RIGHT$(CD$(N%),1)):CHCD%(N%)=VAL(RI
GHT$(CD$(N%),1))
  360 NEXT:ENDPROC
  370 DEF PROCSH:PR1NTTAB(0,0)ST$:COL
OUR1:PRINTTA8(2,0)"- Please wait -":C
OLDUR2:PRINTTA8(2,2)"while I shuffle"
;:VOU7,18,0,0:PRDCCF(1079,421):PROCT(
1):NN%=1:REPEAT
  380 copy=0:J%=1:t%=0:T%=RND(52):RES
TORE:REPEAT:READPK$:t%=t%+1:UNTILt%=T
%:CS$=LEFT$(PK$,1):CN$=RIGHT$(PK$,1)
  390 IFCS$="H"THENPK$=HEART$+CN$
  400 IFCS$="S"THENPK$=SPADE$+CN$
  410 IFCS$="D"THENPK$=DIAMDND$+CN$
  420 IFCS$="C"THENPK$=CLUB$+CN$
  430 REPEAT:CD$(NN%)=PK$:IFJ%=NN%THE
N450
  440 IFCD$(NN%)=CD$(J%)THENcopy=1:UN
TILcopy=1:GOTD380
  450 J%=J%+1:UNTILJ%>13:NN%=NN%+1:UN
TILNN%>13:PRINTTAB(0,2)ST1$:COLOUR3:P
RINTTA8(4,2)"while I deal";:VDU7:PRDC
T(1):ENDPRDC
  460 DATA D9,H4,CJ,ST,CT,D6,S5,S4,CA
,SA,D3,S9,HK,HQ,H8,D8,S6,HA,C9,DT,DK,
S3,SK,H9,H7,DA,SJ,C3,SQ,D5,H3,H2,C4,H
J,H5,D2,C5,HT,C7,S8,H6,C6,C8,S7,C2,D4
,S2,CK,CQ,DQ,D7,DJ
  470 DEF PROCPC(XX%):YY%=214:VDU5:PR
OCcd(p1cdpo%):GCDL0,C%:MOVEXX%+20,YY%
-56:PR1NT;LEFT$(CD$(plcdpo%),1):MOVEX
X%+25,YY%:PRINT;RIGHT$(CD$(p1cdpo%),1
):GCDL0,2:MOVEXX%+25,64:PRINT;p1cdpo%
:pIcdpo%=p1cdpo%+1:VDU4:ENDPRDC
  480 DEF PROC80:PRINTTA8(0,0)ST$:COL
OUR3:PRINTTA8(0,0)"Put 2 cards in cri
b":COLOUR1:VDU7:PRINTTAB(0,2)"Card No
.1 :- ";:INPUTNo1:IFNo1<1ORNo1>6THEN4
80
  490 VDU5,1B,0,0:y%=96:p1x%=((No1*20
0)-200)+80:PROCCF(p1x%,y%):VDU25,4,(p
1x%+80);64;127,4
  500 PRINTTAB(0,2)ST1$:CDLOUR2:VDU7:
PRINTTA8(0,2)"Card No.2 :- ";:INPUTNo
2:1FNo2=No1 DRNo2<1 ORNo2>6THENPRINTT
A8(0,2)STR1NG$(19," "):GOTO500
  510 VDU5,18,0,0:y%=96:p2x%=((No2*20
0)-200)+B0:PROCCF(p2x%,y%):VDU25,4,(p
2x%+80);64;127,4,17,1
  520 Gone%(No1)=99:Gone%(No2)=99:PRI
NTTAB(0,0)ST$:PRINTTAB(0,0)"Please wa
it while I":COLOUR2:PRINT:PRINT" thro
w mine in too";:PROCCP
  530 BDX$(1)=CD$(No1):80X$(2)=CD$(No
2):BDX$(5)=CD$(7):CD$(18)=CD$(7):CD%(
18)=CD%(7):CHCD%(18)=CHCD%(7)
  540 PRINTTAB(0,0)ST$:PRINTTAB(1,1)"
OK, I have chosen.":VDU7,5,18,0,0:y%=
736:c1x%=((NUM1*200)-200)+80:PROCCF(c
1x%,y%):c2x%=((NUM2*200)-200)+80:PRDC
CF(c2x%,y%):PRDCT(1):VDU4:ENDPRDC
  550 DEF PRDCWH:IFdealI%<>-1THENSTART
$=NAME$ ELSESTART$="computer"
  560 PRINTTAB(0,0)ST$:VDU7:COLOUR2:P
RINTTAB(4,1)"Starter card";:PRDCT(1):
VDU5,18,0,3:PRDCST(" "):VDU5,25,4,110
0;536;:IFLEFT$(CD$(7),1)=HEART$ ORLEF
T$(CD$(7),1)=DIAMOND$THENGCDL0,1 ELSE
GCDL0,0
  570 PRINTRIGHT$(CD$(7),1):VDU25,4,1
095;488;:PRINTLEFT$(CD$(7),1):VDU4:PR
INTTA8(0,0)ST$:IFCHCD%(7)=11THENPRDCs
d:PRINTTAB(0,0)ST$:PRINTTA8(0,1)"Two
for his heels!":PRDCT(2):jack%=1
  580 IFdeal%=1ANDjack%=1THENPROCcs(2
) ELSE IFdeal%=TRUE ANDjack%=1THENPRO
Cps(2):jack%=0
  590 VDU7,5,18,0,2,25,4,80;512;:PRIN
T"Game total=":PROCGT:VDU4:ENDPRDC
  600 DEF PROCGT:VDU5,18,0,0,25,4,920
;512;127,127,18,0,1,25,4,800;512;:PRI
NT;totalI%:VDU4:ENDPROC
  610 DEF PRDCSE:A%=B:B%=11:FDRQ%=1TO
3:PROCAL(A%,8%):PROCSW(A%,8%):PRDCCC(
Q%+7,Q%+10):KP%(Q%)=sc%:PRDCAL(A%,8%)
:A%=A%+1:B%=B%+1:NEXT:PRDCKP:IFkp%=0T
HENkp%=RND(3)
  620 IFkp%=1THENBDX$(3)=CD$(12):BDX$
(4)=CD$(13):NUM1=5:NUM2=6:A%=8:B%=11
  630 1Fkp%=2THEN8DX$(3)=CD$(8):BDX$(
4)=CD$(13):NUM1=1:NUM2=6:A%=9:B%=12
  640 IFkp%=3THEN80X$(3)=CD$(8):BDX$(
4)=CD$(9):NUM1=1:NUM2=2:A%=10:B%=13
  650 ENDPROC
  660 DEF PROCT(t):FORT=1TO1500*t:NEX
T:ENDPRDC
  670 DEF PRDCCC(B%,T%):FVR=0:FRR=0:s
c%=0:PROCFR(8%,T%):PRDCTH(B%,T%):PROC
TW(B%,T%):PRDCPR(8%,T%):PROCSM(B%,T%)
:ENDPROC
  680 DEF PRDCFV(B%,T%):LDCALH%,I%,J%
,K%,L%:FORH%=B%TOT%-4:FORI%=H%+1TOT%-
3:FORJ%=I%+1TDT%-2:FDRK%=J%+1TOT%-1:F
ORL%=K%+1TOT%
  690 IFCHCD%(H%)+1=CHCO%(I%)ANDCHCD%
(I%)+1=CHCD%(J%)ANDCHCD%(J%)+1=CHCD%(
K%)ANDCHCD%(K%)+1=CHCD%(L%)THENsc%=sc
%+5:FVR=1:FRR=1
  700 IFCD%(H%)+CD%(I%)+CD%(J%)+CD%(K
%)+CD%(L%)=15THENsc%=sc%+2
  710 NEXT:NEXT:NEXT:NEXT:NEXT:ENDPRO
C
  720 DEF PROCFR(B%,T%):LOCALH%,I%,J%
,K%:FDRH%=B%TDT%-3:FDR1%=H%+1TDT%-2:F
ORJ%=I%+1TOT%-1:FDRK%=J%+1TDT%
  730 IFFVR=0THEN IFCHCD%(H%)+1=CHCD%
(I%)ANDCHCD%(I%)+1=CHCD%(J%)ANDCHCD%(
J%)+1=CHCD%(K%)THENsc%=sc%+4:FRR=1
  740 IFCD%(H%)+CD%(I%)+CD%(J%)+CD%(K
```



```
%)=15THENsc%=sc%+2
  750 NEXT:NEXT:NEXT:NEXT:ENDPROC
  760 DEF PROCTH(B%,T%):LOCALH%,I%,J%
:FORH%=B%TOT%-2:FORI%=H%+1TOT%-1:FORJ
%=I%+1TOT%
  770 IFFRR=0THEN IFCHCD%(H%)+1=CHCD%
(I%)ANDCHCD%(I%)+1=CHCD%(J%)THENsc%=s
c%+3
  780 IFCD%(H%)+CD%(I%)+CD%(J%)=15THE
Nsc%=sc%+2
  790 NEXT:NEXT:NEXT:ENDPROC
  800 DEF PROCTW(B%,T%):LOCALH%,I%:FO
RH%=B%TOT%-1:FOR1%=H%+1TOT%
  810 IFCD%(H%)+CD%(I%)=15THENsc%=sc%
+2
  820 1Ftota1%+CD%(H%)=15THENfiftn%=H
%
  B30 1Ftota1%+CD%(1%)=15THENfiftn%=I
%
  840 NEXT:NEXT:ENDPROC
  850 DEF PROCPR(B%,T%):LOCALH%,I%:FO
RH%=B%TOT%-1:FORI%=H%+1TOT%
  860 1FCHCD%(H%)=CHCD%(1%)THENsc%=sc
%+2:ftpr%=H%
  870 IFCHCD%(H%)=CHCD%(n%)THENpair%=
H%
  880 IFCHCD%(I%)=CHCD%(n%)THENpair%=
I%
  B90 NEXT:NEXT:ENDPROC
  900 DEF PROCSM(B%,T%):same%=0:LOCAL
1%:FORI%=B%TOT%-2:IFLEFT$(CD$(I%),1)=
LEFT$(CD$(I%+1),1)THENsame%=same%+1
  910 NEXT:IFsame%=3THENsc%=sc%+4
  920 IFsame%=3ANDLEFT$(CD$(T%),1)=LE
FT$(CD$(B%),1)THENsc%=sc%+1:ENDPROC E
LSEENDPROC
  930 DEF PROCJK(B%,T%):LOCALI%:FORI%
=B%TOT%-1
  940 IFR1GHT$(CD$(I%),1)="J"ANDLEFT$
(C0$(T%),1)=LEFT$(CD$(I%),1)THENsc%=s
c%+1
  950 NEXT:ENDPROC
  960 DEF PROCsd:FORS=50TO100:SOUND1,
-10,S,.7:NEXT:ENDPROC
  970 DEF PROCSW(b%,t%):LOCALM%,N%:FO
RM%=b%TOt%-1:FORN%=M%+1TOt%:1FCHCD%(M
%)<=CHCD%(N%)THEN990
  980 temp%=CHCD%(M%):CHCD%(M%)=CHCD%
(N%):CHCD%(N%)=temp%
  990 NEXT:NEXT:ENDPROC
 1000 DEF PROCKP:nk%=0:kp%=0:FORI%=1T
O3:1FKP%(I%)>nk%THENnk%=KP%(I%):kp%=I
%
 1010 NEXT:ENDPROC
 1020 DEF PROCGM:PROCAL(A%,B%):PROCCG
(A%,B%):1aid%=1:1Fdeal%=-1THENPROCFM:
laid%=2 ELSEPROCPM:laid%=2:PROCNM:Iai
d%=3
 1030 PROCPM:npair%=0:IFpInogo%=0THEN
laid%=1aid%+1:IFlaid%>3THENPROCrn:IFg
otrun%>=3ANDgotrun%>plcorun%THENPROCp
s(gotrun%):pIcorun%=gotrun%
 1040 IFtotal%=31THENcorp%=2:GOTO1140
 1050 IFcpnogo%=0ANDplnogo%=1THENcorp
%=1:GOTO1070
 1060 IFtotal%<31PROCOD:IFpoint%=1THE
NPROCps(1):point%=0:GOTO1140
 1070 PROCNM:npair%=0:1Fcpnogo%=0THEN
laid%=1aid%+1:IFlaid%>3THENPROCrn:IFg
otrun%>=3ANDgotrun%>plcorun%THENPROCc
s(gotrun%):plcorun%=gotrun%
 1080 1Ftota1%=31THENcorp%=1:GOTO1140
 1090 IFp1%=4ANDcpnogo%=1THEN1060
 1100 IFplnogo%=0ANDcpnogo%=1THENcorp
%=2:GOTO1120
 1110 IFtotal%<31PROCOD:1Fpoint%=1THE
NPROCcs(1):point%=0
 1120 IFplnogo%=0ANDpl%<4THEN1030
 1130 IFcpnogo%=0ANDco%<4THEN1070
 1140 PROCPG:PROCGN:plnogo%=0:cpnogo%
=0:plcorun%=0:1aid%=1:apair%=0:copr%=
0:plpr%=0:CHCD%(n%)=0:npair%=1:total%
=0:PROCGT:FORC%=1TO8:run%(C%)=0:NEXT
 1150 IFpl%=4ANDco%=4THENcorp%=3
 1160 ONcorp% GOTO1030,1070,1170
 1170 PROCPG:PROCGN:COLOUR3:PRINTTAB(
0,0)ST$:PRINTTAB(3,0)"All cards laid"
:COLOUR1:PRINTTAB(5,2);"- scores -":V
DU7:PROCT(2):PRINTTAB(0,0)ST$:plcdpo%
=1:PROCCD(0):GCOL0,0
 1180 PROCCF(p1x%,96):VDU5,25,4,(p1x%
+B0);64;127:PROCCF(p2x%,96):VDU25,4,(
p2x%+80);64;127,4:PROCFC(14,848):PROC
T(1):IFdeal%=1THENPROCPF ELSEPROCCT
 1190 total%=0:PROCBL:PRINTTAB(0,0)ST
$:COLOUR3:PRINTTAB(0,1)"Next game sta
rting";:VDU7:PROCRC:VDU5,18,0,0:PROCC
F(1079,421):VDU4:1Fdeal%=-1THENPROCUP
 ELSEPROCDN
 1200 PROCT(2):deal%=-deal%:ENDPROC
 1210 DEF PROCCG(a%,b%):IFa%=8 c%=6:O
%=13
 1220 IF a%=9 c%=5:O%=12
 1230 IF a%=10 c%=4:O%=11
 1240 FORI%=a%TOb%:CD$(I%+c%)=CD$(I%)
:CD%(1%+c%)=CD%(I%):CHCD%(I%+c%)=CHCD
%(I%):NEXT:ENDPROC
 1250 DEF PROCFM:newno%=newno%+1:COLO
UR1:VDU7:PR1NTTAB(0,0)ST$:PR1NTTAB(1,
1)"Computer to move";:PROCT(1):check%
=0:first%=14:REPEAT:IFCD%(first%)<5TH
ENrun%(laid%)=CHCD%(first%):PROCCL(fi
rst%):check%=1:first%=17
 1260 first%=first%+1:UNTIL first%>17
:IFcheck%=1THENENDPROC
 1270 ftpr%=0:PROCPR(14,17):IFftpr%>0
THENrun%(laid%)=CHCD%(ftpr%):PROCCL(f
tpr%):ENOPROC
 1280 L%=RND(4)+13:run%(laid%)=CHCD%(
L%):PROCCL(L%):ENDPROC
 1290 DEF PROCNM:IFco%=4 OR cpnogo%=1
THENENDPROC
 1300 COLOUR1:VDU7:PR1NTTAB(0,0)ST$:P
RINTTAB(1,1)"Computer to move";:PROCT
(1):newno%=newno%+1:nexcd%=0:fiftn%=0
:pair%=0:thiscd%=cpprecd%:prdrp%=0:IF
tota1%<15THENPROCTW(14,17) ELSE1330
 1310 IFfiftn%<>0THENrun%(laid%)=CHCD
```

→

DEPT. CG1,
18 HAZELMERE RD,
STEVENAGE, HERTS
SG2 8RX.

QUAL - SOFT

STEV (0438) 721936

NEW GENERATION LEISURE SOFTWARE

QUAL-SOFT comments: In 1982/83 it was only necessary to throw together a few M/C code routines, record them on a cassette, produce a flashy cassette insert, and it sold. But as we approach 1985, BBC users have become much more discriminating in their choice of software. Customers demand something more satisfying than a few hours of high adrenaline action. They demand, not hours, but tens of hours, even a hundred hours plus, of "playability". They demand that the program challenges their intellect and not just their reflexes. They demand QUALITY games. For the 84/85 season; for these customers, QUAL-SOFT is proud to offer these games:

"DIVISION ONE '85" and "SOCCER SUPREMO" are direct descendants of the 19B3 game "LEAGUE DIVISION ONE", but they are a significant advance on what was described as "by far the best game for the BBC Micro". They are soccer management SIMULATIONS! Soccer management is about ASSESSMENT; assessment of the opposition strengths, of your players and their skills, of the effectiveness of the teams you assemble, of the value of transfer listed players, of the potential of your own youngsters. Soccer management is about EXPERIMENTATION; experimentation with players, with blends of players, with team formations, with tactical ploys. QUAL-SOFT's unique soccer match simulation, with tackles, passes, shots, saves etc. all simulated within your micro, and the results displayed in graphical form on your screen, allows ASSESSMENT and EXPERIMENTATION to be carried out. **The result is a 2ND GENERATION soccer management SIMULATION, not an exercise in mental arithmetic!!!**

BBC 'B'
(all OS)

DIVISION ONE '85'

TAPE £9.95
DISC £12.95
(all inc.)

This is an uncompromising strategy/tactics simulation, it will test to the full your knowledge of the game and your ability to use it to build a winning side. As described above, your ability to experiment and judge the results are paramount. Injuries, suspensions, long term "form" changes, short term "form" fluctuations, will all conspire to confuse and defeat you. You will also meet the prblems of financial limitations as you choose between the 33 players available to you. You have a contract for 5 seasons, which will be terminated if your side is relegated. This game is the supreme test of your managerial skills, short of the real thing.

ELECTRON
BBC 'B' (all OS)

SOCCER SUPREMO

TAPE £9.95
DISC £12.95
(all inc.)

Soccer Supremo has been improved over the original game in the direction of a much more realistic "3D", 22 MAN, FULL PITCH match simulation for each of your games. The game is still a significant challenge to your managerial skills, but with greater emphasis on enjoyment and entertainment. But with 30 players to choose from, restricted finances, and just a 5 season contract, the 1st Division Championship is not an easy task.

COMBO OFFER. Telephone conversations, and orders, have shown that customers find it very diffcult to choose between the two games. Our COMBO tape/disc offers you both games for just the price of one plus £3. £12.95 or £15.95.
PREVIOUS OWNERS. Send in your ORIGINAL tape/disc of LEAGUE DIVISION ONE, with manuals, and receive £7 for your tape or £9 for your disc, discount against any order for the above game(s).

BBC 'B'
(all OS)

PORTFOLIO

TAPE £9.95
DISC £12.95
(all inc.)

The most popular indoor games have always been multi-player board games, and yet designers have insisted on producing single player "me and my micro" computer games. In the USA the balance has already shifted away from the "arcade mentality" towards multi-player programs. PORTFOLIO is an investment game for 1-4 players, or teams of players. Each player/team begins with £1,000 and invests his money in a variety of companies against a background of continually changing international, national and commercial news. The news items affect the share values of the companies in a variety of different ways, and the players must shift their money around to maximise their investments; maybe even to become a millionaire.
The game is an ideal family game, or perfect in the classroom with a high pupil/micro ratio. It's suitable for 12 years old and above, though younger do enjoy the game with some help from their friends, and in this game, competitors. For the adult we have made some of our news items somewhat cryptic, occasionally even red herrings, to really make them think about the effect on share values. Because we call it a family game we have tried to see that there is something in it for all members of the family. Teachers can even tailor the news items to suit the capabilities of the pupils.
So why not set your Beeb on the coffee table, connect it to the TV set and play against the rest of the family or your friends on these cold, wet winter nights that are on us. It might even bring back the art of conversation (as against BASIC statements).

All orders DIRECT SALE.
Orders by post delivered in 3/4 working days.
ACCESS telephone authorisation 1/2 days.

TAPE		DISC
£9.95 □	DIVISION ONE '85'	□ £12.95
£9.95 □	SOCCER SUPREMO	□ £12.95
£12.95 □	COMBO, DIV.1+SS	□ £15.95
£9.95 □	PORTFOLIO	□ £12.95

Soccer Supremo: ELECTRON or BBC

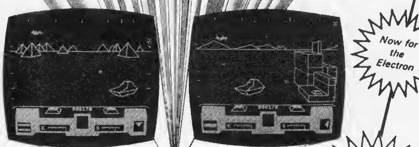
Name: ....
Address: ....
....
....
Card No: ....

From Page 63

```
%(fiftn%):PROCCL(fiftn%):PROCcs(2) EL
SE1330:IFCD$(fiftn%)=CD$(n%)THENPROCc
s(2)
 1320 GOTD1430
 1330 PROCPR(I4,17):1Fpair%<>0THEN 1F
totalI%+CD%(pair%)<=31THENrun%(Iaid%)=
CHCD%(pair%):PRDCCL(pair%):PROCcs(2):
apair%=apair%+1:prdrp%=1ELSEapair%=0
 1340 IFapair%=2THENPROCcs(4)ELSE IFa
pair%=3THENPROCcs(10):apair%=0
 1350 IFprdrp%=1THEN1430
 1360 IFlaid%<=2THEN1400ELSEfindrun%=
0:1p%=14
 1370 REPEAT:IFCHCD%(1p%)<31THENrun%(
laid%)=CHCD%(1p%) ELSE1390
 1380 PRDCrn:1Fgotrun%>=3THEN IFtotal
%+CD%(Ip%)<=31THENrun%(1aid%)=CHCD%(I
p%):PROCCL(1p%):Ip%=17:findrun%=1
 1390 lp%=1p%+1:UNTIL1p%>17:IFfindrun
%=1THEN1430
 1400 FORI%=14TOI7:IFtotal%+CD%(I%)<=
31THENnexcd%=I%:I%=17
 1410 NEXT:IFnexcd%<>0THENrun%(laid%)
=CHCD%(nexcd%):PRDCCL(nexcd%):GOTO143
0
 1420 VDU4,7,18,0,1:PR1NTTAB(0,0)ST$:
PRINTTAB(1,i)"Sorry - can't go!":newn
o%=newno%-1:cpnogo%=1:PRDCT(1):ENDPRD
C
 1430 IFpInogo%=1ANDnpair%=0ANDRIGHT$
(CD$(thiscd%),1)=R1GHT$(CD$(cpprecd%)
,1)THENPROCcs(2):copr%=copr%+I:IFcopr
%=2THENPROCcs(4)
 1440 IFp1%=4ANDnpair%=0ANDpIfin%=1AN
DRIGHT$(CD$(thiscd%),1)=RIGHT$(CD$(cp
precd%),1)THENPRDCcs(2):copr%=copr%+I
:IFcopr%=2THENPROCcs(4)
 1450 IFtota1%=31THENPRDCcs(2):cpnogo
%=I
 I460 IFp1%=4 p1fin%=1
 1470 ENDPROC
 1480 DEF PROCCL(twoetc%):tota1%=tota
1%+CD%(twoetc%):PROCLA(twoetc%,736):C
HCD%(twoetc%)=newno%:CD%(twoetc%)=new
no%:X2%(co%)=twoetc%:cpprecd%=twoetc%
:co%=co%+1:PROCGT:ENDPROC
 1490 DEF PROCPM:IFpI%=4 ORp1nogo%=1T
HENENDPROC
 1500 PR1NTTA8(0,0)ST$:VDU7,17,3:PR1N
TTA8(8-(LEN(NAME$)/2),0)NAME$;:CDLOUR
2:PRINT" to":PR1NTTAB(1,1)"enter card
 number":PRINT" or";:COLOUR1:PRINT"'G
'";:CDLOUR2:PRINT"if can't go";:*FX21
,0
 1510 N$=GET$:IFN$="G"ORN$="g"THENp1n
ogo%=1:ENDPROC
 1520 n%=VAL(N$):IFn%<1 ORn%>6THEN150
0
 1530 IFGone%(n%)=99THENPRINTTAB(0,0)
ST$:COLOUR1:PRINTTAB(0,1)"Sorry-alrea
dy gone";:VDU7:PROCT(2):GOTO1500
 1540 IFtotalI%+CD%(n%)>31THENCOLOUR1:
PR1NTTAB(0,0)ST$:PRINTTA8(0,1)"Sorry-
more than 31";:VDU7:PROCT(2):GOTO1500
 1550 IFco%=0THEN1590
 1560 IFnpair%=0ANDRIGHT$(CD$(n%),I)=
RIGHT$(CD$(X2%(co%-1)),1)THENPROCps(2
):apair%=apair%+1ELSEapair%=0
 1570 IFapair%=2THENPRDCps(4)ELSE IFa
pair%=3THENPROCps(10):apair%=0
 1580 IFco%=4ANOnpair%=0ANOcpfin%=IAN
ORIGHT$(CD$(n%),I)=RIGHT$(CD$(p1precd
%),1)THENPRDCps(2):p1pr%=p1pr%+1:1FpI
pr%=2THENPROCps(4)
 1590 IFcpnogo%=1ANDnpair%=0ANDRIGHT$
(CD$(n%),1)=RIGHT$(CD$(p1precd%),1)TH
ENPRDCps(2):p1pr%=p1pr%+1:IFp1pr%=2TH
ENPRDCps(4)
 1600 IFtota1%+CD%(n%)=15THENPROCps(2
)
 1610 IFtotalI%+CD%(n%)=3ITHENPROCps(2
):cpnogo%=1
 1620 tota1%=totalI%+CD%(n%):PRDCGT:X1
%(pI%)=((n%*200)-200)+80:VDU5,25,4,(X
1%(pI%)+80);64;127,4:p1%=pI%+1:Gone%(
n%)=99:p1precd%=n%:IFco%=4THENcpfin%=
1
 1630 run%(Iaid%)=CHCO%(n%):ENDPROC
 I640 DEF PROCOD:IFcpnogo%=1ANOp1%=4
ORcpnogo%=1ANOp1nogo%=1THENpoint%=1:E
NDPROC
 1650 IFco%=4 ANDp1%=4 ORco%=4 ANDp1n
ogo%=1THENpoint%=1 ELSE ENDPROC
 1660 DEF PROCLA(pos%,YY%)
 1670 XX%=(((pos%-D%)*200)-200)+80:VD
U5,18,0,3:PRDCCF(XX%,YY%):PRDCcd(pos%
):VDU25,4,(XX%+24);(YY%+120);:GCOL0,C
%:PR1NTRIGHT$(CD$(pos%),1):VDU25,4,(X
X%+24);(YY%+54);:PRINTLEFT$(CD$(pos%)
,1):VDU4,I8,0,3:ENDPRDC
 1680 DEF PROCPH:HH%=1:LDCALI%:FORI%=
1TD6:IFI%<>NoI ANDI%<>No2 THENCD$(HH%
)=CD$(I%):HH%=HH%+1
 1690 NEXT:CD$(5)=CD$(7):ENDPROC
 1700 DEF PROCPG:VDU5,18,0,0:LOCALI%:
FDRI%=0TOpI%-1:PRDCCF(X1%(I%),96):NEX
T:VDU4:ENDPROC
 1710 DEF PROCGN:VDU5,18,0,0:LOCALI%:
FORI%=0TOco%-1:x2%=(((X2%(I%)-D%)*200
)-200)+80:PROCCF(x2%,736):NEXT:VDU4:E
NDPROC
 1720 DEF PROCcs(points%):VDU5,18,0,2
:R%=1:REPEAT:IFPOINT(1008,550)=2THENV
OUI8,0,1
 1730 MDVE ABS(CXM%),CYM%:PRINT ".":I
FCXM%=48THENCXM%=-CXM%+32:CYM%=568
 1740 IFA8S(CXM%)=976ANOCYM%=568THEN
CXM%=1008:CYM%=600
 I750 SOUND1,-12,R%*2+50,2:PROCT(.3):
IFPDINT(1008,582)=1THENPRDCcn(1,688)
 1760 IFPDINT(1008,550)=1THENPROCWN("
The computer")
 1770 CXM%=CXM%-32:R%=R%+1:UNTILR%>po
ints%:VDU4:ENDPROC
 17B0 DEF PROCSC(8%,T%):PROCAL(B%,T%)
:PRDCSW(B%,T%):FVR=0:FRR=0:sc%=0:PROC
FV(B%,T%):PROCFR(B%,T%):PRDCTH(8%,T%)
:PRDCTW(B%,T%):PRDCPR(B%,T%):PROCJK(8
%,T%):PROCSM(B%,T%):ENDPROC
 1790 DEF PRDCCD(F%):FORx%=BOTD1080ST
EP200:y%=736:IFF%=0ANDx%=80 DRF%=0AND
x%=1080THENGCOL0,0:PROCCF(x%,y%):GOTO
1810
 IB00 GCDL0,3:PRDCCF(x%,y%)
 1810 IFF%=1THENGCOL0,1:PROCCK(x%,y%)
 1820 GCDL0,3:y%=96:PROCCF(x%,y%):PRD
CPC(x%):NEXT:ENDPROC
 1830 DEF PROCps(score%):VDU5,18,0,2:
R%=1:REPEAT:IFPOINT(1008,422)=2THENVD
U1B,0,1
 1840 MOVEABS(PXM%),PYM%:PR1NT ".":IF
PXM%=48THENPXM%=-PXM%+32:PYM%=440
 1850 IFPXM%=4BTHENPXM%=-PXM%+32:PYM%
=440
 1860 IFABS(PXM%)=976ANDPYM%=440THENP
XM%=1008:PYM%=472
 1870 SOUND2,-12,I00-R%*2,2:PROCT(.3)
:IFPOINT(1008,454)=1THENPROCcn(2,320)
 1880 IFPOINT(1008,422)=1THENPROCWN(N
AME$)
 1890 PXM%=PXM%-32:R%=R%+1:UNTILR%>sc
ore%:VDU4:ENDPRDC
 1900 DEF PRDCFC(IC%,y%):LDCALI%:VDU5
:FORI%=280T0880STEP200:PROCcd(IC%):GC
OL0,C%:MOVE I%+25,y%-56:PRINT;LEFT$(C
D$(IC%),1):MOVE1%+25,y%:PRINT;RIGHT$(
CO$(IC%),1):IC%=IC%+1:NEXT:VDU4:ENOPR
OC
 I910 DEF PROCUP:VDU5,18,0,1,25,4,110
5;320;9,I27,25,4,1105;688;230,7,4:END
PROC
 1920 DEF PRDCON:VDU5,1B,0,2,25,4,110
5;68B;9,127,25,4,1105;320;230,7,4:END
PRDC
 I930 DEF PROCRC:VDU5,18,0,0:FORx%=B0
TO1080STEP200:y%=736:PRDCCF(x%,y%):y%
=96:PROCCF(x%,y%):VDU25,4,(x%+96);64;
127:NEXT:VDU4:ENDPRDC
 1940 DEF PROCWN(W$)
 I950 PROCBL:VDU4,17,I:PRINTTA8(0,0)S
T$:PRINTTA8(10-(LEN(W$)/2),0) W$:CDLD
UR2:PRINT:PRINTTAB(3);"is the winner!
";:TI%=1:VDU5:REPEAT:FORI%=1TO3:VDU1B
,0,I%,25,4,970;512;(226+I%):PROCsd:NE
XT:VDU25,4,1034;512;127
 1960 PROCT(.2):TI%=TI%+1:UNTILTI%=5:
VDU4,17,3:PRINTTA8(0,0)ST$:PR1NTTA8(1
,0)"Would you like":COLOUR2:PRINTTAB(
2);"another game of":COLOUR1:PRINTTA8
```

3D-TANKZONE

Sensational NEW release!!!



Now for the Electron

For BBC 32k

£8·95

SPECIAL OFFER TO READERS
All 5 games + "Brainteasers" on one disc. 40 track or 80 track.
Just £29 00

At last, a stunning interpretation of the 3D Wire-Graphics Tank Battle simulation you've all been waiting for brought to you by Dynabyte.

Features include anti-tank missile launcher with realistic 'shell-in-flight' action, radar screen, detailed control panel, anti-aircraft and anti-tank modes (with options to choose one or both), hidden line removal, exploding tanks and many other special effects. There is NO other game like it.

DYNABYTE
EXPLOSIVE Software

Availiable now at all good computer shops or direct from

DYNABYTE SOFTWARE
PO Box 10 Barnsley S. Yorkshire
0226-287707

POOL £6.95

Exciting pool hall action all the way in this 2-player game of skill. Excellent ball control and smooth graphics an outstanding feature of this program.

BBC and Electron versions (Machine code 32k)

CORPORATE CLIMBER £6.95

Avoid the eager taxmen in the lifts ready to hinder your climb to power and beware of too much stress resulting in high blood pressure.

BBC and Electron versions (Machine code 32k)

HORSERACE £6.95

All the excitement of the track brought to your own home in this colourful game complete with tumbling jockeys, realistic horses, TV van, tote and leader boards, waving crowds and much more. Suitable for 1-6 players.

BBC and Electron versions (Basic + M/C 32k)

LEMMING SYNDROME £6.95

Save the panicking population in this highly original game full of special features and options designed to make your task harder as you get better.

BBC and Electron versions (Machine code 32k)

Disc versions available, same price for Mail Order only

SAE for catalogue.

(All programs require O.S. 1.2)

Please state BBC or Electron when ordering and include 50p P&P

PHILIPS 14" COLOUR TV/MONITOR

£235 inc VAT

TV/Monitor (resolution better than 585 × 450 pixels. Dot-to-dot distance .65mm)

(TV only: £185 inc VAT)

As a company we chose the Philips because we believe it to have the clearest picture as well as being the best value. *Take our word for it, or come and see for yourself.*

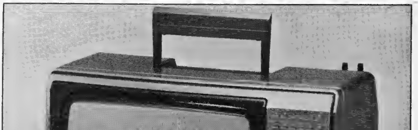
EASY ON-THE-SPOT PARKING. AND IT'S *FREE*!

Viglen COMPUTER SUPPLIES

Plus £8 carriage and packing

Personal Service & Quality coupled with Wholesale Prices,
Unit 7 Trumpers Way, Hanwell, London W7 2QA.
Telephone 01-843 9903

For special attention to your orders or enquiries concerning MONITORS, PRINTERS, COMPUTERS, DISC DRIVES, ring 01-843 9903 NOW or fill in the coupon below.



● Easy to carry ● Lightweight ● Sparkling picture in seconds ● Reliability and performance excellent ● Headphone Socket can also be connected to tape recorder ● Loudspeaker mute switch ● Retractable carrying handle ● Monitor/TV selected by switch ● Includes RGB lead connecting direct to your BBC Micro. ● Philips Model No. 14CT 2006/05T

Post to: VIGLEN COMPUTER SUPPLIES, UNIT 7, TRUMPERS WAY, HANWELL, LONDON W7 2QA. MU 11/7

Please send me ____ (qty) Philips TV/Monitor at £243 each. I enclose Cheque/P.O. for £ ________ made out to

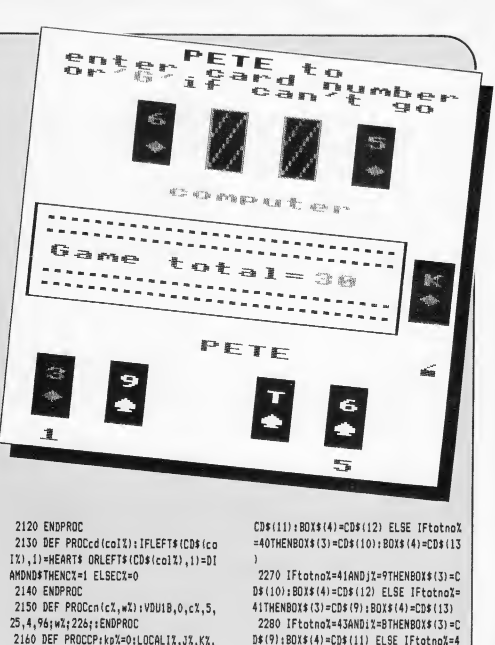
VIGLEN COMPUTER SUPPLIES or debit my ACCESS/BARCLAYCARD No. ________

Name ________ Signature ________

Address ________

From Page 65

```
(3);"cribbage - (Y/N)";:yn$=GET$:IFyn
$<>"Y"THEN END ELSE GOTO70
 1970 DEF PROCPF:VDU7,17,2:PRINTTAB(1
0-(LEN(NAME$)/2),1) NAME$:PROCPH:PROC
SC(1,5):IFsc%>0THENPROCscore:PROCps(s
c%):PROCBL
 1980 PRINTTAB(0,1)ST1$:COLOUR1:PRINT
TAB(6,1)"computer":VDU7:PROCSC(14,18)
:IFsc%>0THENPROCscore:PROCcs(sc%):PRO
CBL
 1990 PROCRC:PROCBX(14,736):PROCSC(14
,18):IFsc%>0THENPROCscore:PROCcs(sc%)
:PROCBL ELSECOLOUR1:PRINTTAB(0,0)ST$:
PRINTTAB(6,0)"B##!#* it":COLOUR2:PRIN
TTAB(2,2)"my box is empty!":PROCsd:PR
OCT(3):ENDPROC
 2000 PROCT(2):ENDPROC
 2010 DEF PROCCT:VDU7,17,1:PRINTTAB(6
,1)"computer":PROCSC(14,18):IFsc%>0TH
ENPROCscore:PROCcs(sc%):PROCBL
 2020 PRINTTAB(0,1)ST1$:COLOUR2:PRINT
TAB(10-(LEN(NAME$)/2),1)NAME$:VDU7:PR
OCPH:PROCSC(1,5):IFsc%>0THENPROCscore
:PROCps(sc%):PROCBL
 2030 PROCRC:PROCBX(1,96):PROCSC(1,5)
 2040 IFsc%>0THENPROCscore:PROCps(sc%
):PROCBL ELSECOLOUR1:PRINTTAB(0,0)ST$
:PRINTTAB(7-(LEN(NAME$)/2),0) "Sorry
";:COLOUR3:PRINTNAME$:COLOUR2:PRINTTA
B(2,1)"There's nothing":PRINTTAB(3);"
in your box!";:PROCsd:PROCT(3):ENDPRO
C
 2050 PROCT(2):ENDPROC
 2060 DEF PROCBX(Z%,Y%):T%=1:LOCALI%:
FORI%=Z%TOZ%+4:CD$(I%)=BOX$(T%):T%=T%
+1:NEXT:VDU5,18,0,3:FORx%=280TO880STE
P200:PROCCF(x%,Y%):NEXT:PROCFC(Z%,Y%+
112):ENDPROC
 2070 DEF PROCMN:PRINTTAB(13,2);CHR$1
34;CHR$141;"CRIBBAGE":PRINTTAB(13);CH
R$134;CHR$141;"CRIBBAGE"
 2080 PRINTTAB(0,7);CHR$133;"NOTE:";C
HR$131;"With the exceptions of enteri
ng":PRINTTAB(6);CHR$131;"your name an
d boxcard selection":PRINTTAB(6);CHR$
131;"all entries are single key."
 2090 PRINTTAB(6);CHR$129;"At the end
 of each game the":PRINTTAB(6);CHR$12
9;"computer will display the scores":
PRINTTAB(6);CHR$129;"You must then pr
ess a key to":PRINTTAB(6);CHR$129;"co
ntinue."
 2100 PRINTTAB(1,17);CHR$134;"Now ent
er your name-(1-14 letters)":PRINT:IN
PUT LINETAB(10)NAME$
 2110 IFLEN(NAME$)>14ORNAME$=""THENPR
INTTAB(0,17);STRING$(199," "):VDU7:PR
INTTAB(0,16);CHR$136;CHR$129;"Come on
, please enter correctly!";:PROCT(4):
GOTO2100
 2120 ENDPROC
 2130 DEF PROCcd(col%):IFLEFT$(CD$(co
l%),1)=HEART$ ORLEFT$(CD$(col%),1)=DI
AMDND$THENC%=1 ELSEC%=0
 2140 ENDPROC
 2150 DEF PROCcn(c%,w%):VDU18,0,c%,5,
25,4,96;w%;226;:ENDPROC
 2160 DEF PROCCP:kp%=0:LOCALI%,J%,K%,
L%:FORI%=8TO10:FORJ%=I%+1TD11:FORK%=J
%+1TO12:FORL%=K%+1TO13:totno%=0:CD$(1
4)=CD$(I%):CD$(15)=CD$(J%):CD$(16)=CD
$(K%):CD$(17)=CD$(L%)
 2170 PROCAL(14,17):PROCSW(14,17):PRO
CCC(14,17):IFsc%>kp%THENi%=I%:j%=J%:k
%=K%:l%=L%:totno%=i%+j%+k%+l%:PROCCX:
kp%=sc%
 2180 NEXT:NEXT:NEXT:NEXT
 2190 IFkp%=0THENPROCSE:ENDPROC
 2200 CD$(14)=CD$(i%):CD$(15)=CD$(j%)
:CD$(16)=CD$(k%):CD$(17)=CD$(l%):CD$(
9)=CD$(14):CD$(10)=CD$(15):CD$(11)=CD
$(16):CD$(12)=CD$(17):NUM1=1:NUM2=6:A
%=9:B%=12:ENDPROC
 2210 DEF PROCCX:IFtotno%=38THENBOX$(
3)=CD$(12):BOX$(4)=CD$(13)
 2220 IFtotno%=39THENBOX$(3)=CD$(11):
BOX$(4)=CD$(13)
 2230 IFtotno%=42THENBOX$(3)=CD$(8):B
OX$(4)=CD$(13)
 2240 IFtotno%=45THENBOX$(3)=CD$(8):B
OX$(4)=CD$(10)
 2250 IFtotno%=46THENBOX$(3)=CD$(8):B
OX$(4)=CD$(9)
 2260 IFtotno%=40ANDk%=10THENBOX$(3)=
CD$(11):BOX$(4)=CD$(12) ELSE IFtotno%
=40THENBOX$(3)=CD$(10):BOX$(4)=CD$(13
)
 2270 IFtotno%=41ANDj%=9THENBOX$(3)=C
D$(10):BOX$(4)=CD$(12) ELSE IFtotno%=
41THENBOX$(3)=CD$(9):BOX$(4)=CD$(13)
 2280 IFtotno%=43ANDi%=8THENBOX$(3)=C
D$(9):BOX$(4)=CD$(11) ELSE IFtotno%=4
3THENBOX$(3)=CD$(8):BOX$(4)=CD$(12)
 2290 IFtotno%=44ANDi%=8THENBOX$(3)=C
D$(9):BOX$(4)=CD$(10) ELSE IFtotno%=4
4THENBDX$(3)=CD$(8):BOX$(4)=CD$(11)
 2300 ENDPRDC
 2310 DEF PROCBL:VDU5,18,0,0,25,4,103
2;512;127,127,127,127,127,127,127,127
,127,127,127,127,127,127,127,4:ENDPRO
C
 2320 DEF PROCscore:PROCBL:VDU5,18,0,
2,25,4,80;512;:PRINT"Total Points=";:
VDU18,0,1:PRINT;sc%:VDU4:get=GET:*FX2
1,0
 2330 ENDPROC
 2340 DEF PROCrn:LOCALI%,J%,K%:gotrun
%=1:FORI%=1TOlaid%-2:FORJ%=I%+1TOlaid
%-1:IFrun%(I%)<=run%(J%)THEN2360
 2350 run_temp%=run%(I%):run%(I%)=run
%(J%):run%(J%)=run_temp%
 2360 NEXT:NEXT:FORK%=1TOlaid%-1:IFru
n%(K%)+1=run%(K%+1)THENgotrun%=gotrun
%+1ELSE IFrun%(K%+1)<>0THENgotrun%=1
 2370 NEXT:ENDPROC
 2380 IFERR=25THEN70ELSEMODE7:REPORT:
PRINT" at line No. ";ERL:VDU23;11,255
;0;0;0
```



PETE to
enter card number
or 'G' if can't go

computer

Game total=30

PETE

Spend a night on the tiles...

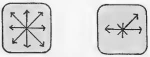
. . . but make sure you head in the right direction with JERRY LANCASTER's game of strategy for two players

THE principle of Which-Way? was described in 1969 by a research worker looking into psycho-cybernetics and game playing. Each of your moves is partly controlled by your opponent's previous move, which means there is an element of feedback in the game.

In the same way, your move controls the options open to your opponent, so you may be able to force him into a move which is favourable to yourself.

The board is made up of 64 tiles, set out in the usual 8 × 8 pattern. The two opposing sides are Red and White. Each side has a start tile and a finish tile and the other 60 are made up of arrow tiles or blank tiles.

Each arrow tile shows three arrows and since there are eight possible directions for the arrows this means there will be 56 different possibilities for the arrow tiles.

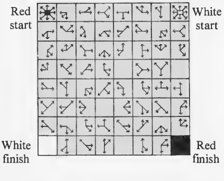


8 directions   one possible tile

$$\frac{8 \times 7 \times 6}{1 \times 2 \times 3} = 56$$

If we take a completely blank tile into account then there will be 57 possible tiles to choose from. The program chooses at random from these designs and prints them out in the 8 × 8 layout.

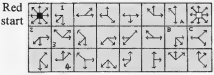
On the first run this takes about five or six seconds to be produced, but in further games the board is printed on demand.



The aim of the game is to get from Red start to Red finish or from White start to White finish.

There is one basic rule which controls the moves in the game – you can only move in a direction shown by one of the arrows on your opponent's tile. To make your moves you use the keys shown in Figure I.

Here is part of one possible layout showing the start of a game. Red moves first. He has three choices, tiles 1, 2 or 3:



If Red moves to tile 1, White has a completely free choice – A, B or C.

White chooses A, Red will have only one legal move (down to tile 3) because the other two directions take him off the board.

White chooses B, Red has a choice of three moves all of which take him in the right direction.

White chooses C, Red will be left with two moves, one of which takes him back to the Red start tile, giving White a free choice next move.

If Red moves to tile 2, White will have only two possible moves, B or C.

White chooses B, and gives Red the advantage of a free choice of three directions.

White chooses C, and Red has two moves possible this time, back up to tile 1 or down to tile 4.

If Red moves to tile 3, White has no legal moves available because all Red's arrows take White off the board.

This move by Red to tile 3 looks like his best option.

If you attempt an illegal move the computer will not execute it. Instead it will leave you where you are and award the move to your opponent.

Illegal moves are:

- Any move which would take you off the board.
- Any move which does not match one of the arrows on your opponent's tile.
- Any move which would put both of you onto the same tile.

Some tiles are blank. If you move onto a blank tile your opponent will be left with no legal moves, so in effect you will get a free move. When you are forced to stay on the same tile you can do so by pressing the central key of your control group.

Red always starts the game but White can decide whether or not to accept the layout displayed. If Y is pressed you can get an entirely new layout, without losing the current game score.

This option is also available during play, so that if a drawn position is reached you can restart that game.

Be careful when entering the program that you don't inadvertently include any spaces or suchlike as it is very tight on memory. You will certainly have to download it on DFS machines.

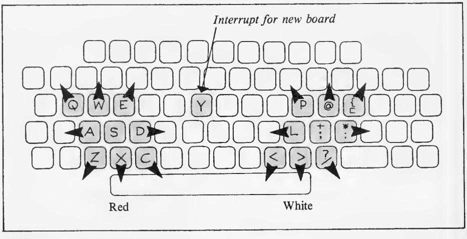
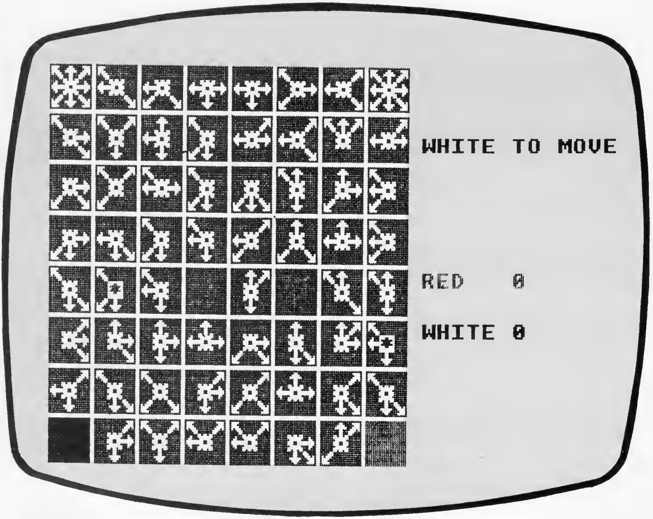
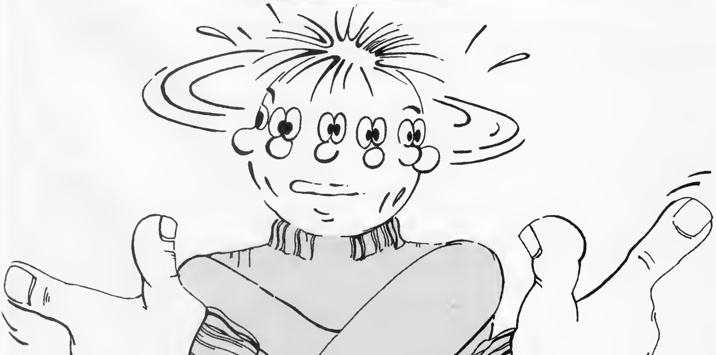


Figure I



```
WHITE TO MOVE

RED    0

WHITE 0
```



From Page 68

PROCEDURES AND MAIN VARIABLES

DEFPROCinit

sh$(N%) This is an entire tile pattern made up from nine user defined characters and several cursor moves, all printed as one unit.

SHAPE%(N%) A Binary coded nine-digit number which describes the tile pattern.

TILE%(X,Y) Array which records the layout of the tiles on the board for a particular game.

red, white The game scores.

VDU23,224 to 232 'Arrow' shapes.

VDU23,234 to 242 'No arrow' shapes.

DEFPROCtileshapes

move1$, move 2$ Cursor movement controls which form part of each string describing a tile.

SHAPE%(N%) Read as a number from data statements. The data is written in the form 111010000.

This is derived from a tile pattern:

1	1	1
0	1	0
0	0	0

SH$ String version of SHAPE%(N%). Leading zeros will have been omitted and need to be replaced.

sh$(N%) The tile pattern obtained from SH$. MID$(....) is used to detect the '1' which shows where the arrow should be.

DEFPROCboard

X% Y% These correspond to the columns and rows of the 8 × 8 board.

Values are used to identify particular tiles, such as the corners. The remaining tile patterns are chosen at random using N% = RND(57).

RED and WHITE are printed on their start squares using "*" in the right colours.

DEFPROCgame

PROCmoveA
PROCmoveB
I% Used to move each 'man'.

Uses INKEY(0) to detect key presses. Acceptable returns are converted into values of I% from 1 to 9. I% then corresponds to tile sectors 1 to 9.

1	2	3
4	5	6
7	8	9

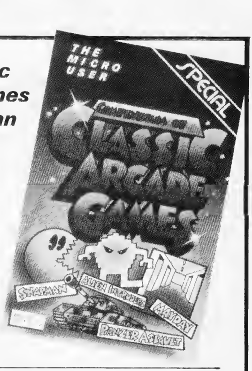
PROClegalmove The values are checked using MID$(...) to determine the legality of a move.

PROCwin If a winning position is reached a winning message is printed in the appropriate colour and the game score is updated.

```
   20 REM (C) The Micro User
   30 MODE1
   40 VDU23,1,0;0;0;0;
   60 *FX4,1
   70 *FX11,0
   90 PROCinit
  100 PROCtileshapes
  110 PROCboard
  120 PROCgame
  130 COLOUR2:CLS:GOTO110
  140 END
  160 DEFPROCinit
  165 ENVELOPE2,2,6,0,0,255,0,0,126,0
,0,-126,126,126
  170 DIMsh$(58)
  180 DIMTILE%(8,8)
  190 DIMSHAPE%(58)
  200 red=0:white=0
  210 PRINTTAB(7,7);"WHICH-WAY?"
  220 VDU23,224,0,127,67,71,67,81,120
,124
  230 VDU23,225,0,255,231,195,129,231
,231,231
  240 VDU23,226,0,254,194,226,194,138
,30,62
  250 VDU23,227,127,119,103,64,64,103
,119,127
  260 VDU23,228,36,0,129,24,24,129,0,
36
  270 VDU23,229,254,238,230,2,2,230,2
38,254
  280 VDU23,230,124,120,81,67,71,67,1
27,0
  290 VDU23,231,231,231,231,129,195,2
31,255,0
  300 VDU23,232,62,30,138,194,226,194
,254,0
  310 VDU23,234,0,127,127,127,127,127
,127,127
  320 VDU23,235,0,255,255,255,255,255
,255,255
  330 VDU23,236,0,254,254,254,254,254
,254,254
  340 VDU23,237,127,127,127,127,127,1
27,127,127
  350 VDU23,238,255,255,255,255,255,2
55,255,255
  360 VDU23,239,254,254,254,254,254,2
54,254,254
  370 VDU23,240,127,127,127,127,127,1
27,127,0
  380 VDU23,241,255,255,255,255,255,2
55,255,0
  390 VDU23,242,254,254,254,254,254,2
54,254,0
  400 PRINTTAB(8,14);"Setting up boar
d"
  410 ENDPROC
  420 DEFPROCtileshapes
  430 move1$=CHR$(8)+CHR$(8)+CHR$(8)+
CHR$(10)
  440 move2$=CHR$(11)+CHR$(11)
  450 FORN%=1TO58
  460 READSHAPE%(N%)
  470 SH$=STR$(SHAPE%(N%))
  480 REPEAT
  490 IF LEN((SH$))<9 SH$="0"+SH$
  500 UNTIL LEN((SH$))=9
  510 sh$(N%)=""
  520 FORA%=1TO9
  530 IF MID$(SH$,A%,1)="1"sh$(N%)=sh
$(N%)+CHR$(223+A%)ELSEsh$(N%)=sh$(N%)
+CHR$(233+A%)
  540 IFA%=3ORA%=6sh$(N%)=sh$(N%)+mov
e1$
  550 IFA%=9sh$(N%)=sh$(N%)+move2$
  560 NEXTA%
  570 NEXTN%
  580 ENDPROC
  590 DATA111010000,011110000,0011110
```

WHAT A BARGAIN!

Four classic arcade games for less than £1.50 each



Three of this high-powered collection are top-rate machine-code versions of arcade classics and the fourth is a thrilling real-time adventure game. There's hours of enjoyment and something to suit everyone in this unique value for money collection.

ALIEN INTRUDERS – With only your laser for protection you must destroy the waves of aliens who threaten to engulf you. **PANZER ATTACK** – You are a tank commander engaged in vicious combat against encircling enemy forces. **SNAPMAN** – Guide your man through the maze as he munches energy pellets and avoids hostile aliens. **MAYDAY** – A futuristic adventure! As captain of an interstellar cruiser you must guide the sole survivor of a stricken space freighter through the wreckage of his craft.

£5.95 (cassette) **£7.95** (disc)

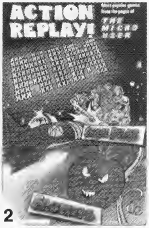
ACTION REPLAY!

Build your own handy, action-packed cassette library of these all-time Micro User favourites. Only £3.75 per cassette (three games in each).

Deathwatch Hold the fort against a combined tank and helicopter assault on your positions Fast and furious – one of our most popular games

Pelmanism Defeat the computer by remembering more pairs of cards than it can

King Kong Use your helicopter to rescue two maidens from the Empire State building – avoiding a deluge of rocks from King Kong



Fruities Fast paced, fun-packed extravaganza Guide your man up ladders, avoiding – and digging traps for – the marauding Fruities!

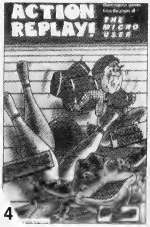
Nim Find out just how nimble your mind is in this intriguing battle of wits against the computer.

Space Pilot Desperately fighting off hostile life-forms, you must dock with satellites to replenish dwindling fuel supplies

Robin and Marian Medieval melodrama! Rescue Maid Marian from the sheriff A hilarious, action-packed game

EECGAL In charge of the EEC's Picture Gallery, you have to put some order into the frames – but it's not so simple as it sounds

Air Strike You need to fly fast and low for this dangerous intrusion into enemy air space



Barrel Battles Fast and furious video action Dodging an avalanche of deadly barrels you climb towards the telephone

Frogs 'n' Dogs Loved by young and old alike Simply take Freddy Frog for a walk avoiding Fido and poisonous lilies!

Tenpins Stunning simulation of ten pin bowling This addictive game will hold you spellbound

Tower of London Ghosts have broken out of the Tower of London and taken the treasure Can you recover it?

Blackjack A true-to-life recreation of the casino favourite Can you beat the bank?

Galactic Invaders The only thing between you and imminent destruction by waves of hostile aliens is your proton gun A compulsive classic

Alien Intruders Stunning version of the all-time arcade hit No one can resist this classic!

Goblins Frolics in fairyland! Guide Kaplan the pixie home through the forests, avoiding the nasty goblins – a delightful game

Gone Fishing Superb simulation of a day on the riverside This enthralling program will give hours of enjoyment

Please send me:

CLASSIC ARCADE GAMES

BBC 'B' Cassette £5.95	□
40 Track Disc £7.95	□
80 Track Disc £7.95	□

ACTION REPLAY

BBC 'B' Cassette £3.75	1	□	4	□
	2	□	5	□
	3	□	6	□

□ I enclose cheque made payable to Databasa Publications Ltd. for £ ______

□ I wish to pay by □ Access □ Visa

No. ______________________ Expiry date ______

Signed ______________________

Name ______________________

Addrass ______________________

____

Post to: Database Publications, 68 Chester Road,
Hazel Grova, Stockport SK7 5NY.

GS1

From Page 70

```
00,000111100,000011110,000010111
  600 DATA110110000,110011000,1100101
00,110010010,110010001
  610 DATA011011000,011010100,0110100
10,011010001
  620 DATA001110100,001110010,0011100
01,101110000
  630 DATA000111010,000111001,1001110
00,010111000
  640 DATA000011101,100011100,0100111
00,001011100
  650 DATA100010110,010010110,0010101
10,000110110
  660 DATA100010011,010010011,0010100
11,000110011,000011011
  670 DATA101011000,101010100,1010100
10,101010001
  680 DATA100110100,100110010,1001100
01
  690 DATA100011010,100011001,1000101
01
  700 DATA010110100,010110010,0101100
01
  710 DATA010011010,010011001,0100101
01
  720 DATA001011010,001011001,0010101
01,000110101
  730 DATA000000000,111111111
  740 DEFPROCboard
  750 X%=1
  760 Y%=1
  770 CLS
  780 COLOUR2
  790 Z%=0
  800 REPEAT
  810 N%=RND(57)
  820 IFX%=1ANDY%=1N%=58
  830 IFX%=8ANDY%=1N%=58
  840 IFX%=1ANDY%=8 N%=57
  850 IFX%=8ANDY%=8 N%=57
  860 TILE%(X%,Y%)=SHAPE%(N%)
  870 PRINTsh$(N%);
  880 X%=X%+1
  890 IFX%>8PRINT:PRINT:PRINT:X%=1:Y%
=Y%+1
  900 IFX%=1ANDY%=8 COLOUR3 ELSE COLO
UR2
  910 IFX%=8ANDY%=8 COLOUR1
  920 UNTILY%>8
  930 XA=1:YA=1:XB=22:YB=1
  940 COLOUR1:PRINTTAB(XA,YA);"*"
  950 COLOUR3:PRINTTAB(XB,YB);"*"
  960 *FX21,0
  970 ENDPROC
  980 DEFPROCgame
  990 REPEAT
 1000 COLOUR1:PRINTTAB(27,4);"RED TO
MOVE  "
 1010 PRINTTAB(27,12);"RED   "red
 1020 COLOUR3:PRINTTAB(27,15);"WHITE
"white
 1030 PROCmoveA
 1040 SOUND1,-15,97,2
 1050 IFZ%=1GOTO1090
 1060 COLOUR3:PRINTTAB(27,4);"WHITE T
O MOVE"
 1070 PROCmoveB
 1080 SOUND1,-15,197,2
 1090 UNTILZ%=1
 1100 IFXA=22ANDYA=22red=red+1
 1110 IFXB=1ANDYB=22white=white+1
 1120 COLOUR1:PRINTTAB(27,12);"RED
"red
 1130 COLOUR3:PRINTTAB(27,15);"WHITE
"white
 1140 PRINT TAB(4,28)"Press space for
 next game"
 1150 REPEAT UNTIL GET$=" "
 1160 ENDPROC
 1170 DEFPROCmoveA
 1180 REPEAT
 1190 xa=XA:ya=YA
 1200 I%=INKEY(0)
 1210 UNTILI%<>-1
 1220 IF I%=81 I%=1
 1230 IF I%=87 I%=2
 1240 IF I%=69 I%=3
 1250 IF I%=65 I%=4
 1260 IF I%=83 I%=5
 1270 IF I%=68 I%=6
 1280 IF I%=90 I%=7
 1290 IF I%=88 I%=8
 1300 IF I%=67 I%=9
 1310 IFI%=89 Z%=1:ENDPROC
 1320 IF I%<1 OR I%>9 THEN SOUND1,-15
,33,7:GOTO1180
 1330 PROClegalmove(XB,YB)
 1340 IFI%=-1 ENDPROC
 1350 IF I%=1 XA=XA-3:YA=YA-3
 1360 IF I%=2 YA=YA-3
 1370 IF I%=3 XA=XA+3:YA=YA-3
 1380 IF I%=4 XA=XA-3
 1390 IF I%=5 XA=XA:YA=YA
 1400 IF I%=6XA=XA+3
 1410 IF I%=7 XA=XA-3:YA=YA+3
 1420 IF I%=8 YA=YA+3
 1430 IF I%=9 XA=XA+3:YA=YA+3
 1440 IFXA<1 XA=xa:YA=ya
 1450 IFXA>22 XA=xa:YA=ya
 1460 IFYA<1 XA=xa:YA=ya
 1470 IFYA>22 XA=xa:YA=ya
 1480 IFXA=XB AND YA=YB XA=xa:YA=ya
 1490 COLOUR2:PRINTTAB(xa,ya);CHR$(22
8)
 1500 COLOUR1:PRINTTAB(XA,YA);"*"
 1510 PROCcheck
 1520 ENDPROC
 1530 DEFPROCmoveB
 1540 REPEAT
 1550 I%=INKEY(0)
 1560 UNTILI%<>-1
 1570 xb=XB:yb=YB
 1580 IF I%=80 I%=1
 1590 IF I%=64 I%=2
 1600 IF I%=91 I%=3
 1610 IF I%=76 I%=4
 1620 IF I%=59 I%=5
 1630 IF I%=58 I%=6
 1640 IF I%=44 I%=7
 1650 IF I%=46 I%=8
 1660 IF I%=47 I%=9
 1670 IF I%=89 Z%=1:ENDPROC
 1680 IF I%<1 OR I%>9 THEN SOUND1,-15
,33,7:GOTO1540
 1690 PROClegalmove(XA,YA)
 1700 IFI%=-1 ENDPROC
 1710 IF I%=1 XB=XB-3:YB=YB-3
 1720 IF I%=2 YB=YB-3
 1730 IF I%=3 XB=XB+3:YB=YB-3
 1740 IF I%=4 XB=XB-3
 1750 IF I%=5 XB=XB:YB=YB
 1760 IF I%=6 XB=XB+3
 1770 IF I%=7 XB=XB-3:YB=YB+3
 1780 IF I%=8 YB=YB+3
 1790 IF I%=9 XB=XB+3:YB=YB+3
 1800 IFXB<1 XB=xb:YB=yb
 1810 IFXB>22 XB=xb:YB=yb
 1820 IFYB<1 XB=xb:YB=yb
 1830 IFYB>22 XB=xb:YB=yb
 1840 IF XB=XA AND YB=YA XB=xb:YB=yb
 1850 COLOUR2:PRINTTAB(xb,yb);CHR$(22
8)
 1860 COLOUR3:PRINTTAB(XB,YB);"*"
 1870 PROCcheck
 1880 ENDPROC
 1890 DEFPROCcheck
 1900 IFXA=22 AND YA=22 PROCwin
 1910 IFXB=1 AND YB=22 PROCwin
 1920 ENDPROC
 1930 DEFPROCwin
 1935 SOUND1,2,4,50
 1940 PRINTTAB(27,4);"              "
 1950 PRINTTAB(4,25);"The winner ! *
* * *"
 1960 Z%=1
 1970 ENDPROC
 1980 DEFPROClegalmove(X,Y)
 1990 x=(X+2)/3:y=(Y+2)/3
 2000 a$="0"
 2010 Z$=STR$(TILE%(x,y))
 2020 REPEAT
 2030 IF LEN(Z$)<9 THEN Z$=a$+Z$
 2040 UNTILLEN(Z$)=9
 2050 IF MID$(Z$,I%,1)<>"1" I%=-1
 2060 ENDPROC
```

The best software value ever!

£7.95
Cassette/Disc

£14.00
Both Discs

IT'S an offer you can't afford to refuse – all 14 fantastic games from The Micro User's Games Special are now available on tape or disc to save you the trouble of typing them in.

Never before has such high quality fun-packed games been offered at such a low price!

In fact the package is so crammed with goodies that for the DFS version we've had to split the cassette over two discs.

On DISC A (Cassette Side A) you'll find Bounce, Crib, Fair Play, Sevens, Which Way, White Widow and Yams.

DISC B (Cassette Side B) features China Drop, Crown Green, Draw Poker, Explorer Eddie, Masterspy, Patience and Splat.

And every game is of the high standard for which Micro User is renowned!

Never before has so much BBC Micro software been offered for so little.

Don't miss out – send off for your cassette or discs today

Please send me Games Special software as indicated here:

□ I enclose cheque made payable to Database Publications Ltd.
for £ ____________

□ I wish to pay by □ Access □ Visa

No. ____________ Expiry date ____________

□ Cassette £7.95
□ Disc A £7.95
□ Disc B £7.95
□ Discs A and B £14.00
□ 40 track □ 80 track
N B No Electron version

Signed ____________

Name ____________

Address ____________

____

Post to: Games Special, Database Publications,
68 Chester Road, Hazel Grove, Stockport SK7 5NY.

Dicey business

Who can get the best score from throwing a set of dice? Play Yams by JULIA AIZPURVS and find out!

YAMS is an adaptation of the traditional dice game Yahtzee and can be played by two to four players.

The object of the game is to get the highest score possible by using a set of five dice to obtain certain combinations, called scores. These are outlined below.

Each player takes a turn at throwing the set of dice. During that turn you have a maximum of three throws. After the first throw you may

□ *Throw all the dice again.*

□ *Hold selected disc and throw the rest.*

□ *Accept the throw.*

After the third throw, you must accept the dice shown. If you can select a score from this combination of dice then you must do so – if not you can pass.

Once you have made a valid selection, a summary of your current scores is shown and the game then passes on to the next player. It cycles round all the players until everyone has had 14 turns – enough to cover all the scores.

A final score sheet is then shown and the winner declared.

KEYS	
Y	*Accept a throw*
Space	*Throw all dice again*
H	*Hold some of the dice*

Once you have pressed H, you are asked how many dice you wish to keep and then which ones they are. To select these, type in the appropriate number (1-5) of the dice, reading from left to right. Once selected, the dice has the letter H printed under it.

When you accept a throw, type the NAME of the score you have selected. The ones you require are shown at the top of the screen. If you can't make up a score, type PASS and then enter the NAME of the score you wish to forfeit.

When your score summary is shown the computer waits for quite a while to allow you to get a good look at your scores.

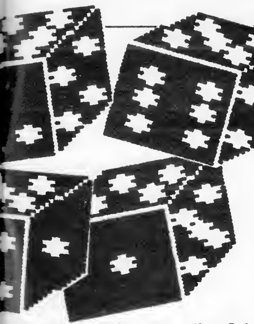
As the game progresses, there are

VARIABLES

A$,AB$,AN$	Strings for INPUTS.
A%	Accept throw flag.
a%	Number of selected 'score'.
B%	Bonus.
b%, B$	General variable used in PROCPASS.
C%	Game loop counter.
d%, dd%	General loop variable.
DR$	Names of players who have drawn a game.
D$	General string used in PROCgot.
e%	Side value in PROCSIDE.
f%	Indicates whether or not you've already selected a score.
F%, ff%	General variable in RUN procedure.
FA%	Value of face of dice.
G%	General loop variable.
g%	Face value in PROCFACE.
H%	General loop variable.
HO$, HO%	Number of dice to be held.
ho%	Identity of dice selected to hold.
I%	Seeds random number generator.
ID%	Flag to indicate validity of throw.
J%	General loop variable.
J$	General string used in PROCPASS.
k%	General loop variable.
K%	Key used in sort routine.
N%	General loop variable.
0%	General variable in RUN procedure.
P%	Number of players.
p%	General loop variable.
pts%	Points assigned to a score.
q%	Controls colours.
Q%, R%	General loop variable.
REPLY%	Indicates a key pressed.
S%	Size of side of dice.
s%	General loop variable.
ss%	General variable used in PROCRUN.
SI%	Value of side of dice.
throw%	Throw counter.
TT%, t%	General variables in PROCRUN.
T%	Controls dice moving down screen.
t%	Value of top of dice in PROCTOP.
to%	Value of top of dice.
U%	General loop variable.
V%	Valid flag.
WIN$	Name of winning player.
X%	Value of dice in PROCTHDI.
X%	Coordinate (x) of dice.
Y%	Coordinate (y) of dice.
Z%	Delay variable.

ARRAYS

D%	Holds value for each of the five dice.
DPOS%	Holds starting position for drawing dice on screen.
hold%	Contains dice to be held during each throw.
NA$	Holds names of players.
SC%	Holds each players total score.
SC$	Holds names of scores.
SCSH%	Three dimensional array holding scoresheet for each player. That is, SCSH%(A,B,C) where A='score', B=player and C=status of 'score' A for player B. C=0 gives the actual score obtained; C=1 gives details about that score, that is, whether available, forfeited or taken.



quite a lot of them. However, if you find this too long, simply alter line 1190 PROCDEL (1000) to a lower value.

When playing this game, you take your scores from the front face of the dice – that is, the dice facing you on the screen.

Because of the large number of variables in this game, the program had to be written in Mode 4, thus sacrificing colours. The program is a tight fit even then, so be careful not to leave spaces.

SCORES

Ones	Any number of ones. Score total number of ones (1-5).
Twos	Any number of twos. Score total number of twos (2-10).
Threes	Any number of threes. Score total number of threes (3-15).
Fours	Any number of fours. Score total number of fours (4-20).
Fives	Any number of fives. Score total number of fives (5-25).
Sixes	Any number of sixes. Score total number of sixes (6-30).

If you total of scores is 60 or more, you are awarded a bonus score of 60 points.

SPECIAL SCORES

Two Pairs	Any two pairs of dice, such as 2 ones and 2 fives. Score 30 points.
Triple	Any three dice the same. Score 30 points.
Run	All dice different, either one to five or two to six. Score 40 points.
Four Kind	Four of dice the same. Score 40 points.
Full House	Two dice the same, and the other three the same as each other, such as 2 fours and 3 sixes. Score 40 points.
Hi	Any fairly high score. It must be higher than Lo. Score total number shown on dice face.
Lo	Any low score. It must be lower than Hi. Score total number shown on dice face.
Yams	Five sixes. Score 60 points.

PROCEDURES

Lines 10-70	Main body of program. Seeds random number generator, defines envelopes and calls procedures to play the game. Turns off cursor after each mode change.
PROCnoplayers	Prints titles. Asks if you want instructions and if so calls PROCINST. Asks for the number of players.
PROCINIT	Sets up arrays according to the number of players and initialises variables.
PROCNAMES	Gets the names of the players.
PROCPLAY	The main procedure, it controls the playing of the game. It calls PROCTHDI, PROCgot, PROCPASS, PROCCHECK, PROCerr, PROCDEL and PROCSCORES.
PROCCHECK	Checks if the score you've selected is valid. Calls relevant checking procedure.
PROCtot	If a number score is selected, this procedure is called to add up the dice with the number.
PROChi	Checks the dice to see if a Hi score is valid.
PROClo	Checks the dice to see if a Lo score is valid.
PROCvalid	Checks SCSH% after validation. If there is no score for the player concerned, then ID% is returned false, that is not valid.
PROCTPR	Checks all combinations of dice for a Two Pairs score.
PROCTRI	Checks dice for a Triple score.
PROCFKD	Checks dice for a Four Kind score.
PROCYAMS	Checks dice for a Yams score.
PROCRUN	Checks dice for a Run score.
PROCFH	Checks dice for a Full House score.
PROCSCORES	Prints the summary of scores at the end of each players turn. Indicates scores obtained and scores forfeited.
PROCDEL	Delay procedure.
PROCADD	Adds up each players scoresheet and decides whether to award bonus. Calls PROCPLTBL.
PROCPLTBL	Draws the final scores table and declares the winner. Asks if you want another game.
PROCTHDI	Throws dice, checking which ones (if any) are to be held. This procedure assigns the values to the dice.
PROCDRDI	Actually draws dice tumbling down the screen. Also erases dice where necessary. Sets hold% back to 0 before ending.
PROCPLOT	Draws a cube at selected position.
PROCLPLOT	Draws the edges of the cube.
PROCTILT	Draws a tiltedcube and its edges.
PROCTOP	Draws the spots on the top of the cube.
PROCSIDE	Draws the spots on the side of the cube.
PROCFACE	Draws the spots on the face of the cube.
PROCSPOTS	Calculates the values for the top and side of the cube given the value for the face.
PROCHOLD	Asks which dice you wish to hold and sets hold% accordingly.
PROCPASS	Asks which score you will forfeit and sets SCSH% to indicate this.
PROCERASE	Erases dice from resting position on screen, ready for next throw.
PROCgot	Checks to see if you've selected a score which you've already had.
PROCINST	Instructions for the game.
PROCerr	Lets you know if you've entered a name or score that the program does not recognise.
Line 2580	Data containing names of scores.

From Page 75

```
   10  REM *****YAMS*****
   20  REM BY J.AIZPURVS
   30  REM (C) The Micro User
   40 MDDE7:VDU23;8202;0;0;0;:I%=RND(
-T1ME):PROCnoplayers:PRDC1NIT:PRDCNAM
ES:MODE4:VDU23;8202;0;0;0;24,0;160;12
79;840;19,0,4,0,0,0:CLS:CLG
   50 ENVELOPE1,1,-128,127,-128,1,1,1
,75,-50,-1,0,126,0
   60 PROCPLAY:MDDE7:VDU23;8202;0;0;0
;:PRDCADD
   70 END
   80 DEFPRDCnoplayers
   90 PRINTTAB(12,8);CHR$141;CHR$136;
CHR$133;CHR$157;CHR$132;"YAMS ";CHR$3
2;CHR$156;CHR$137:PR1NTTAB(12,9);CHR$
141;CHR$136;CHR$133;CHR$157;CHR$132;"
YAMS ";CHR$32;CHR$156;CHR$137:*FX21,0
  100  PRINTTAB(3,14);CHR$131;"Do you
 want instructions?";CHR$134;"Y/N":AN
$=GET$
  110 IF AN$="Y"THEN PRDC1NST
  120 *FX21,0
  130 PRINTTAB(3,14);SPC(37);TAB(3,14
);CHR$130;:1NPUT"HOW MANY PLAYERS ",P
%
  140  IFP%<=1 OR P%>4THENSDUNDO,-15,
3,10:SOUND1,0,172,10:GDTD130
  150  ENDPROC
  160 DEFPRDCINIT
  170 DIM D%(4),SCSH%(14,P%-1,1),NA$(
P%-1),DPOS%(4),SC$(14),SC%(P%-1),hold
%(4)
  180 S%=100
  190  FOR N%=0TO4:hold%(N%)=FALSE:NE
XT
  200 FOR N%=0TD4:D%(N%)=0:NEXT
  210 FOR N%=0TDP%-1:FDR p%=0TO14:FDR
Q%=0TO1:SCSH%(p%,N%,Q%)=0:NEXT:NEXT:N
EXT
  220 FOR d%=0TO4:READ DPOS%(d%):NEXT
  230 DATA 400,600,800,1000,1200
  240 RESTORE2580:FDR N%=1TO14:READ S
C$(N%):NEXT
  250 FOR N%=0TOP%-1:SC%(N%)=0:NEXT
  260  ENDPROC
  270 DEFPROCNAMES
  280 FDR N%=0TDP%-1:*FX21,0
  290 PRINTTAB(0,17);SPC(40):PR1NTTAB
(3,16);CHR$129;"WHAT 1S YDUR NAME PLA
YER ";N%+1:INPUT,AN$:1F AN$="" THEN 2
90
  300 NA$(N%)=LEFT$(AN$,6):NEXT
  310 ENDPROC
  320 DEFPRDCPLAY:C%=0:REPEAT:C%=C%+1
  330 FORN%=0TOP%-1:CLS:PRINTTAB(5,0)
;"D.K.";NA$(N%);" you need:"
  340 FOR Q%=1TO14:IFSCSH%(Q%,N%,1)=0
THENPR1NTSC$(Q%);",";
  350 NEXT:A%=FALSE:throw%=0
  360 REPEAT:PROCTHDI:UNTIL A%=TRUE O
R throw%=3
  370 *FX21,0
  380 PRINTTAB(0,24);SPC(160):1NPUTTA
B(5,24)"What will you take from this
throw ",AN$:1F LEFT$(AN$,4)="PASS" TH
EN PRDCPASS:GOTO430
  390  f%=0:PROCgot(AN$):1Ff%<>0 THEN
380
  400 ID%=FALSE:PROCCHECK(AN$)
  410 IF1D%=FALSE AND a%=0 THEN PROCe
rr:PROCDEL(500):GOTO380 ELSEIFID%=FAL
SE THEN PRINTTAB(5,26);AN$;" is not v
alid with this throw":SDUNDO,-15,3,10
:SDUND1,0,172,10:PRDCDEL(500):GOTD370
  420 PR1NTTAB(0,24);SPC(160):PRINTTA
B(5,24);"O.K.":SCSH%(a%,N%,1)=1:VDU7:
PROCDEL(100)
  430 PR1NTTAB(0,0);SPC(200);TAB(0,24
);SPC(160):PROCSCORES(NA$(N%)):NEXT
  440 UNT1L C%>=14
  450 ENDPROC
  460 DEFPROCCHECK(A$):a%=0
  470 IFA$="ONES"THENa%=1:PROCtot:PRO
Cvalid
  480 IFA$="TWOS"THENa%=2:PROCtot:PRO
Cvalid
  490 1FA$="THREES"THENa%=3:PROCtot:P
ROCvalid
  500 1FA$="FOURS"THENa%=4:PROCtot:PR
OCvalid
  510 1FA$="F1VES"THENa%=5:PROCtot:PR
OCvalid
  520 IFA$="SIXES"THENa%=6:PROCtot:PR
OCvalid
  530 1FA$="TWD PA1RS"THENa%=7:pts%=3
0:PROCTPR:PROCvalid
  540 1FA$="TR1PLE"THENa%=8:pts%=30:P
ROCTRI:PROCvalid
  550 1FA$="RUN"THENa%=9:pts%=40:PRDC
RUN:PROCvalid
  560 1FA$="FOUR K1ND"THENa%=10:pts%=
40:PRDCFKD:PROCvalid
  570 IFA$="FULL HOUSE"THENa%=11:pts%
=40:PROCFH:PROCvalid
  580 IFA$="H1"THENa%=12:PROChi:PRDCv
alid
  590 IFA$="LO"THENa%=13:PRDClo:PROCv
alid
  600 1FA$="YAMS"THENa%=14:pts%=60:PR
OCYAMS:PRDCvalid
  610  1F a%=0 THEN ID%=FALSE
  620  ENDPROC
  630 DEFPROCtot:J%=0
  640 FOR R%=0TO4:IF D%(R%)=a% THEN J
%=J%+D%(R%)
  650  NEXT:SCSH%(a%,N%,0)=J%
  660  ENDPROC
  670  DEFPROChi
  680 s%=0:1Fa%=12AND SCSH%(13,N%,0)=
0THENFORR%=0TD4:SCSH%(a%,N%,0)=SCSH%(
a%,N%,0)+D%(R%):NEXT ELSEFDRR%=0TO4:s
%=s%+D%(R%):NEXT
  690 IFs%=0THEN ENDPRDC ELSEIFs%<=SC
SH%(13,N%,0)THEN ENDPROC ELSESCSH%(a%
,N%,0)=SCSH%(a%,N%,0)+s%:ENDPRDC
  700  DEFPRDClo
  710 s%=0:1FSCSH%(12,N%,0)=0THENFORR
%=0TD4:SCSH%(a%,N%,0)=SCSH%(a%,N%,0)+
D%(R%):NEXT ELSEFORR%=0TO4:s%=s%+D%(R
%):NEXT
  720 1Fs%=0THEN ENDPROC ELSE IFs%>=S
CSH%(12,N%,0)THEN ENDPRDC ELSESCSH%(a
%,N%,0)=SCSH%(a%,N%,0)+s%:ENDPRDC
  730  DEFPRDCvalid
  740 IFSCSH%(a%,N%,0)=0THEN1D%=FALSE
 ELSEID%=TRUE
  750 ENDPRDC
  760 DEFPROCTPR:V%=FALSE
  770 1F(D%(0)=D%(1)AND(D%(2)=D%(3)DR
 D%(2)=D%(4)DR D%(3)=D%(4)))OR(D%(0)=
D%(2)AND(D%(1)=D%(3)DR D%(1)=D%(4)OR
D%(3)=D%(4)))OR(D%(0)=D%(3)AND(D%(1)=
D%(2)DR D%(1)=D%(4)OR D%(2)=D%(4)))TH
ENV%=TRUE
  780 1F(D%(0)=D%(4)AND(D%(1)=D%(2)DR
 D%(1)=D%(3)OR D%(2)=D%(3)))OR(D%(1)=
D%(2)AND D%(3)=D%(4))OR(D%(1)=D%(3)AN
D D%(2)=D%(4))OR(D%(1)=D%(4)AND D%(2)
=D%(3))THENV%=TRUE
  790 IFV%=TRUE THEN SCSH%(a%,N%,0)=p
ts%
  800 ENDPROC
  810 DEFPRDCTR1:V%=FALSE
  820 1F(D%(0)=D%(1)ANDD%(0)=D%(2))OR
(D%(1)=D%(2)ANDD%(1)=D%(3))OR(D%(2)=D
%(3)ANDD%(2)=D%(4))DR(D%(3)=D%(4)ANDD
%(3)=D%(0))OR(D%(4)=D%(0)AND D%(4)=D%
(1))THEN V%=TRUE
  830 IF(D%(0)=D%(1)ANDD%(0)=D%(3))OR
(D%(1)=D%(2)ANDD%(1)=D%(4))OR(D%(0)=D
%(2)ANDD%(0)=D%(3))DR(D%(0)=D%(2)ANDD
%(0)=D%(4))OR(D%(1)=D%(3)ANDD%(1)=D%(
4))OR(D%(1)=D%(3)ANDD%(1)=D%(0))THEN
V%=TRUE
  840 IFV%=TRUE THEN SCSH%(a%,N%,0)=p
ts%
  850 ENDPROC
  860  DEFPRDCFKD:V%=FALSE
  870 IF(D%(0)=D%(1)ANDD%(0)=D%(2)AND
D%(0)=D%(3))OR(D%(1)=D%(2)ANDD%(1)=D%
(3)ANDD%(1)=D%(4))OR(D%(2)=D%(3)ANDD%
(2)=D%(4)ANDD%(2)=D%(0))DR(D%(3)=D%(4
)ANDD%(3)=D%(0)ANDD%(3)=D%(1))DR(D%(4
)=D%(0)ANDD%(4)=D%(1)ANDD%(4)=D%(2))T
HENV%=TRUE
  880 1FV%=TRUE THEN SCSH%(a%,N%,0)=p
ts%
```

```
  890 ENDPROC
  900 DEFPROCYAMS
  910 IFD%(0)=6ANDD%(1)=6ANDD%(2)=6AN
DD%(3)=6ANDD%(4)=6THENSCSH%(a%,N%,0)=
pts%
  920 ENDPROC
  930 DEFPROCRUN:O%=0:TT%=0:t%=0:F%=0
:ff%=0:ss%=0
  940 FORR%=0TO4:IFD%(R%)=1THENO%=O%+
1
  950 NEXT:IFO%>1THENENDPROC
  960 FORR%=0TO4:IFD%(R%)=2THENTT%=TT
%+1
  970 NEXT:IFTT%<>1THENENDPROC
  980 FORR%=0TO4:IFD%(R%)=3THENt%=t%+
1
  990 NEXT:IFt%<>1THENENDPROC
 1000 FORR%=0TO4:IFD%(R%)=4THENF%=F%+
1
 1010 NEXT:IFF%<>1THENENDPROC
 1020 FORR%=0TO4:IFD%(R%)=5THENff%=ff
%+1
 1030 NEXT:IFff%<>1THENENDPROC
 1040 FORR%=0TD4:IFD%(R%)=6THENss%=ss
%+1
 1050 NEXT:IF(O%=1ANDss%<>0)OR(D%=0AN
Dss%<>1)THENENDPROC
 1060 SCSH%(a%,N%,0)=pts%:ENDPROC
 1070 DEFPROCFH:V%=FALSE
 1080 IF(D%(0)=D%(1)ANDD%(0)=D%(2)AND
D%(3)<>D%(0)ANDD%(3)=D%(4))DR(D%(1)=D
%(2)ANDD%(1)=D%(3)ANDD%(0)<>D%(1)ANDD
%(0)=D%(4))OR(D%(2)=D%(3)ANDD%(2)=D%(
4)ANDD%(0)<>D%(2)ANDD%(0)=D%(1))THEN
V%=TRUE
 1090 IF(D%(3)=D%(4)ANDD%(3)=D%(0)AND
D%(1)<>D%(0)ANDD%(1)=D%(2))OR(D%(4)=D
%(0)AND D%(4)=D%(1)ANDD%(2)<>D%(4)AND
D%(2)=D%(3))DR(D%(0)=D%(1)ANDD%(0)=D%
(3)ANDD%(2)<>D%(0)ANDD%(2)=D%(4))THEN
 V%=TRUE
 1100 IF(D%(1)=D%(2)ANDD%(1)=D%(4)AND
D%(0)<>D%(1)ANDD%(0)=D%(3))OR(D%(0)=D
%(2)ANDD%(0)=D%(3)ANDD%(1)<>D%(0)ANDD
%(1)=D%(4))DR(D%(0)=D%(2)ANDD%(0)=D%(
4)ANDD%(1)<>D%(1)ANDD%(1)=D%(3))THEN
V%=TRUE
 1110 IF(D%(0)=D%(2)ANDD%(0)=D%(4)AND
D%(1)<>D%(0)ANDD%(1)=D%(3))DR(D%(1)=D
%(3)ANDD%(1)=D%(4)ANDD%(0)<>D%(1)ANDD
%(0)=D%(2))DR(D%(1)=D%(3)ANDD%(1)=D%(
0)ANDD%(2)<>D%(1)ANDD%(2)=D%(4))THEN
V%=TRUE
 1120 IFV%=TRUE THEN SCSH%(a%,N%,0)=p
ts%
 1130 ENDPROC
 1140 DEFPROCSCORES(NA$(N%)):PRINTTAB
(0,0);SPC(200)
 1150 PRINTTAB(0,0);"O.K.";NA$(N%);"
Your scores so far are:":FORR%=1TO14:
IFSCSH%(R%,N%,1)=1THENPRINT;SC$(R%);"
,";SCSH%(R%,N%,0);",";
 1160  NEXT
 1170 PRINTTAB(5,24);SPC(160);TAB(5,2
4);"Your forfeits:":PRINTTAB(5,25);:F
DRR%=1TO14:IFSCSH%(R%,N%,1)=-1THEN PR
INT;SC$(R%);",";
 1180 NEXT
 1190 PROCDEL(1000):CLS:ENDPROC
 1200 DEFPROCDEL(Z%):TIME=0:REPEAT:UN
TIL TIME=Z%:ENDPROC
 1210 DEFPROCADD:B%=60
 1220 FORG%=0TOP%-1:FORH%=1TO14:SC%(8
%)=SC%(G%)+SCSH%(H%,G%,0)
 1230 IFH%=6AND SC%(G%)>=60THENSC%(G%
)=SC%(G%)+B%:SCSH%(0,G%,0)=60:SCSH%(0
,G%,1)=1
 1240 NEXT:NEXT:PROCPLTBL:ENDPRDC
 1250 DEFPROCPLTBL
 1260 PRINTTAB(11,0);CHR$(141);CHR$13
3;CHR$(136);"FINAL SCDRES";CHR$137:PR
INTTAB(11,1);CHR$(141);CHR$133;CHR$(1
36);"FINAL SCORES";CHR$137
 1270 FORR%=1TDP%:PRINTTAB(0,3);CHR$(
135);CHR$157;TAB(5+R%*7,3);CHR$(128+R
%);NA$(R%-1):NEXT
 1280 FORG%=1TO6:FORR%=1TOP%:PRINTTAB
(0,G%+3);CHR$(135);CHR$(157);CHR$133;
SC$(G%);TAB(7+R%*7,G%+3);CHR$(128+R%)
;SCSH%(G%,R%-1,0):NEXT:NEXT
 1290 FORR%=1TDP%:PRINTTAB(0,10);CHR$
135;CHR$157;CHR$133;"BONUS";TAB(7+R%*
7,10);CHR$(128+R%);SCSH%(0,R%-1,0):NE
XT
 1300 FORG%=7TO14:FORR%=1TOP%:PRINTTA
B(0,8%+4);CHR$135;CHR$157;CHR$133;SC$
(G%);TAB(7+R%*7,G%+4);CHR$(128+R%);SC
SH%(G%,R%-1,0):NEXT:NEXT
 1310 FORR%=1TOP%:PRINTTAB(0,G%+4);CH
R$135;CHR$157;CHR$133;"TOTAL";TAB(7+R
%*7,G%+4);CHR$(128+R%);SC%(R%-1):NEXT
:PROCWIN:ENDPROC
 1320 DEFPROCWIN:WIN$="":DR$="":K%=0:
FDRR%=1TOP%-1:IFSC%(K%)<SC%(R%)THENK%
=R%
 1330 NEXT:FDRR%=0TOP%-1:IFR%=K%THEN1
350
 1340 IFSC%(R%)=SC%(K%)THENDR$=DR$+NA
$(R%)+" and "
 1350 NEXT:IFDR$<>""THENDR$=DR$+NA$(K
%):PRINTCHR$133;"It's a draw between"
;CHR$136;CHR$134;DR$;CHR$137 ELSEWIN$
=NA$(K%):PRINTTAB(5);CHR$133;CHR$136;
WIN$;CHR$137;CHR$134;"wins this game"
 1360 *FX21,0
 1370 PRINTCHR$130;"Do you want to pl
ay again?":A$=GET$:IF A$="Y"THENRUN
 1380 ENDPRDC
```

→

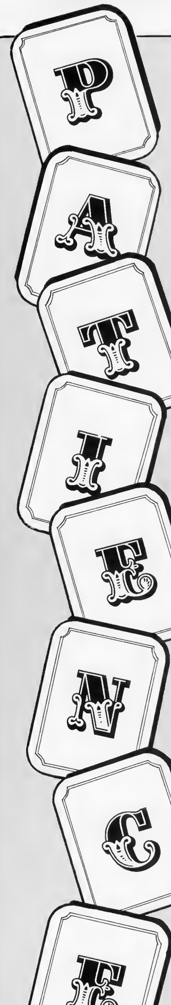
From Page 77

```
 1390 DEFPRDCTHDI:throw%=throw%+1:FOR
dd%=0TO4:1F hold%(dd%)=TRUE THEN NEXT
 ELSE X%=RND(6):D%(dd%)=X%:NEXT
 1400  PROCDRDI:*FX21,0
 1410 PRINTTAB(10,6);"Y = ACCEPT this
 throw";TAB(10,7);"H = HOLD some dice
";TAB(6,8);"SPACE = THROW all dice ag
ain"
 1420 REPLY%=GET:1FREPLY%=32THENA%=FA
LSE ELSEIFREPLY%=89THENA%=TRUE ELSE I
FREPLY%=72THEN A%=FALSE:IFthrow%<>3 T
HENPROCHOLD ELSE SOUND0,-15,3,10:SOUN
D1,0,172,10:8DTO1420
 1430 PR1NTTAB(10,6);SPC(22);TA8(10,7
);SPC(19);TAB(6,8);SPC(29):ENDPROC
 1440 DEFPROCDRD1:IFthrow%=1THEN1450
ELSEFORp%=0TO4:IFhold%(p%)=TRUE THEN
NEXT ELSE PROCERASE(p%):NEXT
 1450 FORp%=0TO4:IFhold%(p%)=TRUE THE
N 1560
 1460 Y%=750
 1470 FORT%=0TD5:FDRq%=1TO0STEP-1:6CO
L0,q%
 1480 X%=DPOS%(p%)-T%*50:VDU29,X%;Y%;
:PROCPLOT:6COL0,0:PROCLPLOT:IFq%=1 TH
EN 6COL0,q%
 1490 8OUND1,1,2,5
 1500  NEXT
 1510 FORq%=1TO0 STEP-1:6COL0,q%
 1520 X%=DPOS%(p%)-T%*50:VDU29,X%;Y%;
:PROCTILT:NEXT:Y%=Y%-50
 1530 NEXT
 1540 X%=DPDS%(p%)-300:VDU29,X%;Y%;:6
COL0,1:PROCPLOT:6COL0,0:PROCLPLOT:PRO
CSPOTS(p%)
 1550 SOUND1,1,2,5:*FX21,5
 1560  NEXT
 1570 FDRk%=0TD4:hold%(k%)=FALSE:PRIN
TTA8(k%*6+3,22);" ";:NEXT:ENDPROC
 1580 DEFPROCPLDT
 1590 MOVE0,S%:DRAWS%,S%:PLOT85,S%/2,
S%/2:DRAW-S%/2,S%-S%/2:PLDT85,0,S%:DR
AWS%,0:DRAWS%,S%:PLOT85,S%/2,S%/2:DRA
WS%/2,-S%/2:PLDT85,S%,0:DRAW-S%/2,S%-
S%/2:DRAWS%/2,S%/2:PLOT85,-S%/2,-S%/2
:DRAWS%/2,-S%/2:PLOT85,S%/2,S%/2
 1600 ENDPROC
 1610 DEFPROCLPLOT
 1620 MDVE-S%/2,S%-S%/2:DRAWS%/2,S%/2
:DRAWS%,S%:MDVES%/2,S%/2:DRAWS%/2,-S%
/2:ENDPROC
 1630 DEFPROCTILT
 1640 MOVE-40,0:DRAW-140,0:DRAW-75,-7
5:DRAW25,-75:DRAW85,0:DRAW25,75:DRAW-
75,75:DRAW-140,0:MOVE-40,0:PLOT85,25,
75
 1650 PLOT85,85,0:MDVE-140,0:MDVE-75,
75:PLOT85,25,75:MOVE85,0:MOVE25,-75:P
LOT85,-40,0:PLOT85,-75,-75:PLOT85,-14
0,0:6COL0,0
 1660 DRAW-20,0:DRAW25/2,-(75/2):MOVE
-20,0:DRAW25/2,75/2:IF q%=1 THEN 8COL
0,q%
 1670  ENDPROC
 1680 DEFPROCTOP(t%):VDU5
 1690 IFt%=1THENMOVE10,87:VDU42
 1700 IFt%=2THENMOVE-30,80:VDU42:MOVE
45,95:VDU42
 1710 IFt%=3THENMDVE10,86:VDU42:MOVE-
32,80:VDU42:MOVE44,95:VDU42
 1720 IFt%=4THENMOVE-25,80:VDU42:MOVE
45,97:VDU42:MDVE0,97:VDU42:MOVE27,80:
VDU42
 1730 IFt%=5THENMOVE-27,80:VDU42:MOVE
47,100:VDU42:MOVE-2,100:VDU42:MOVE29,
80:VDU42:MOVE11,89:VDU42
 1740 IFt%=6THENMOVE-29,76:VDU42:MOVE
54,102:VDU42:MOVE9,102:VDU42:MOVE23,7
6:VDU42:MOVE-12,89:VDU42:MOVE36,89:VD
U42
 1750 VDU4:ENDPROC
 1760 DEFPROCS1DE(e%):VDU5
 1770 IFe%=1THENMOVE61,34:VDU42
 1780 IFe%=2THENMOVE49,48:VDU42:MOVE7
4,17:VDU42
 1790 IFe%=3THENMOVE73,70:VDU42:MOVE4
8,-5:VDU42:MOVE60,32:VDU42
 1800 IFe%=4THENMOVE49,45:VDU42:MOVE7
4,70:VDU42:MOVE49,-5:VDU42:MOVE74,20:
VDU42
 1810 IFe%=5THENMOVE48,47:VDU42:MOVE7
3,68:VDU42:MOVE48,-3:VDU42:MOVE73,18:
VDU42:MOVE60,32:VDU42
 1820 IFe%=6THENMOVE48,45:VDU42:MDVE7
3,70:VDU42:MOVE48,20:VDU42:MOVE73,45:
VDU42:MDVE48,-5:VDU42:MDVE73,20:VDU42
 1830 VDU4:ENDPROC
 1840 DEFPROCFACE(g%):VDU5
 1850 1Fg%=1THENMOVE-11,12:VDU42
 1860 IFg%=2THENMDVE-38,37:VDU42:MOVE
13,-14:VDU42
 1870 IFg%=3THENMOVE-38,37:VDU42:MOVE
13,-14:VDU42:MOVE-11,12:VDU42
 1880 IFg%=4THENMDVE-38,37:VDU42:MOVE
13,-15:VDU42:MDVE-38,-15:VDU42:MDVE13
,37:VDU42
 1890 IFg%=5THENMDVE-38,37:VDU42:MDVE
13,-15:VDU42:MOVE-38,-15:VDU42:MOVE13
,37:VDU42:MDVE-11,12:VDU42
 1900 IFg%=6THENMDVE-38,37:VDU42:MOVE
13,-15:VDU42:MOVE-38,-15:VDU42:MOVE13
,37:VDU42:MDVE-38,12:VDU42:MOVE13,12:
VDU42
 1910 VDU4:ENDPRDC
 1920 DEFPROCSPOTS(N%):8COL0,0:FA%=D%
(N%):IFFA%<>6THEN1950
 1930 to%=RND(5):IFto%=1THEN1930
 1940 IFto%=2THENSI%=3 ELSEIFto%=3THE
NSI%=5ELSEIFto%=4THENSI%=2 ELSESI%=4
 1950 IFFA%<>5THEN1980
 1960 to%=RND(6):IFto%=FA% DRto%=2THE
N1960
 1970 IFto%=1THENSI%=4 ELSE1Fto%=4THE
NSI%=6 ELSE1Fto%=6THENS1%=3 ELSESI%=1
 1980 IFFA%<>4THEN2010
 1990 to%=RND(6):1Fto%=FA% ORto%=3THE
N1990
 2000 1Fto%=1THENSI%=2ELSEIFto%=2THEN
SI%=6 ELSEIFto%=6THENS1%=5 ELSESI%=1
 2010 IFFA%<>3THEN2040
 2020 to%=RND(6):IFto%=FA% ORto%=4THE
N2020
 2030 1Fto%=1THENS1%=5 ELSEIFto%=2THE
NSI%=1 ELSE1Fto%=5THENS1%=6 ELSESI%=2
 2040 IFFA%<>2THEN2070
 2050 to%=RND(6):IFto%=FA% ORto%=5THE
N2050
 2060 IFto%=1THENSI%=3 ELSEIFto%=3THE
NSI%=6 ELSEIFto%=4THENS1%=1 ELSESI%=4
 2070 IFFA%<>1THEN2100
 2080 to%=RND(5):IFto%=FA% THEN2080
 2090 IFto%=2THENS1%=4 ELSEIFto%=3THE
NSI%=2 ELSEIFto%=4THENSI%=5 ELSES1%=3
 2100 X%=DPOS%(N%)-300:VDU29,X%;Y%;:6
COL0,0:PROCTOP(to%):PROCS1DE(SI%):PRO
CFACE(FA%):ENDPROC
 2110 DEFPROCHOLD:*FX21,0
 2120 PR1NTTA8(0,24);SPC(160);TAB(5,2
4);"How many dice do you want to hold
?":HO$=GET$:HO%=VAL(HO$)
 2130 IFHO%<1 OR HO%>5 THEN SOUND0,-1
5,3,10:SOUND1,0,172,10:6OTO2120 ELSE
VDU7
 2140 FORU%=1TOHO%:*FX21,0
 2150 PRINTTAB(0,24);SPC(160);TA8(5,2
4);"Which dice would you like to keep
?":ho%=GET:PR1NTTAB(0,24);SPC(160):VD
U7:IF ho%<49 OR ho%>53 THEN 2150
 2160 hold%(ho%-49)=TRUE:PRINTTAB((ho
%-49)*6+3,22);"H";:NEXT
 2170 ENDPROC
 2180 DEFPROCPASS:RESTORE2580:J$="":b
%=0:*FX21,0
 2190 PRINTTA8(0,24);SPC(160):INPUTTA
8(5,24)"What will you forfeit",8$
 2200  REPEAT:READJ$:b%=b%+1:UNTILJ$=
8$
 2210  SCSH%(b%,N%,1)=-1
 2220 PRINTTA8(0,24);SPC(160):ENDPRDC
 2230 DEFPROCERASE(a%)
 2240 X%=DPOS%(a%)-300:VDU29,X%;Y%;:6
CDL0,0:PROCPLDT
 2250  ENDPROC
 2260  DEFPROCgot(D$):RESTORE2580
 2270  J$="":b%=0
 2280  REPEAT:READJ$:b%=b%+1:UNTILJ$=
D$ OR b%=14:IF b%=14 AND J$<>D$ THEN
ENDPROC
```

```
 2290  IF SCSH%(b%,N%,1)<>0 THEN f%=S
CSH%(b%,N%,1):PRINTTAB(0,24);SPC(160)
;TAB(5,24);"You've already had this o
ne":SOUND0,-15,3,10:SOUND1,0,172,10:P
ROCDEL(300)
 2300  ENDPROC
 2310  DEFPROCINST:*FX21,0
 2320  CLS:PRINTTAB(3,3);"This gaee i
s for 2 to 4 players.You  have five d
ice and you eust use thee to accrue s
cores (details later).You eust  also
atteept to get your 'nueber' scoresto
 a total of 60 or eore to obtain your
 Bonus."
 2330  PRINT;"Player one throws first
.You have three  throws,and after the
 first throw, you   eay hold any nueb
er of dice.To do this, INPUT how eany
 dice you want to hold,andthen which
ones they are (1-5) reading  left to
right."
 2340  PRINT;"Keep CAPS LOCK on durin
g this gaee, as  DATA is stored in up
per case."''''TAB(5);CHR$130;CHR$136;
"PRESS ANY KEY TO CONTINUE.....":AB$=
GET$:CLS
 2350 PRINTTAB(3,3);"To";CHR$129;"ACC
EPT";CHR$135;"a throw press Y"'TAB(3)
;"To";CHR$130;"REJECT";CHR$135;"a thr
ow press SPACE BAR"'TAB(3);"To";CHR$1
31;"HOLD";CHR$135;"any dice press H"
 2360  PRINT;"Once you've accepted yo
ur throw or had  three goes, the Coep
uter asks you to    type in the score
 you wish to take."
 2370  PRINT;"The Coeputer will then
check your Answerto see if it is vali
d.If so,the score   will be calculate
d and added to your    scoresheet.The
 gaee then passes on to"
 2380 PRINT;"the next player,and cont
inues going     round until all the s
cores are filled."'"You also have the
 option to";CHR$134;"PASS"'"To do thi
s type PASS when you are asked what y
ou want to take froe the throw,"
 2390  PRINT;"then decide which score
 you will forfeitThese are displayed
at the bottoe of thescreen."
 2400  PRINTTAB(5);CHR$133;CHR$136;"P
RESS ANY KEY TO CONTINUE.....";:AB$=G
ET$:CLS
 2410  PRINTTAB(8,3);CHR$141;CHR$135;
CHR$157;CHR$132;"nuaber SCORES";CHR$3
2;CHR$156;TAB(8,4);CHR$141;CHR$135;CH
R$157;CHR$132;"nuaber SCORES";CHR$32;
CHR$156
 2420  PRINTTAB(3,7);CHR$129;"ONES...
.points depend on nueber of"'CHR$129;
"ones,froe 1 - 5.";TAB(3,10);CHR$130;
```

```
"TWOS....as above,points froe 2 - 10.
"
 2430  PRINTTAB(3,12);CHR$131;"THREES
..as above,points froe 3 - 15.";TAB(3
,14);CHR$133;"FOURS..as above,points
froe 4-20."
 2440  PRINTTAB(3,16);CHR$134;"FIVES.
..as above,points froe 5-25.";TAB(3,1
8);CHR$129;"SIXES..as above,points fr
oe 6-30.";TAB(5,24);CHR$135;CHR$157;C
HR$132;"PRESS ANY KEY TO CONTINUE....
"CHR$32;CHR$156;:AB$=GET$:CLS
 2450 PRINTTAB(8,3);CHR$141;CHR$135;C
HR$157;CHR$132;"special SCORES";CHR$3
2;CHR$156;TAB(8,4);CHR$141;CHR$135;CH
R$157;CHR$132;"special SCORES";CHR$32
;CHR$156
 2460  PRINTTAB(3,6);CHR$130;"TWO PAI
RS..of 5 dice,2 pairs i.e 2"'CHR$130;
"ones and 2 fives.Points 30.";TAB(3,8
);CHR$131;"TRIPLE..any three dice the
 saee."'CHR$131;"Points 30.";TAB(3,10
);CHR$133;"RUN....Each dice different
, 1-5 or"
 2470 PRINTCHR$133;"2-6, points 40.";
TAB(3,12);CHR$134;"FOUR KIND..Any fou
r dice the saee."'CHR$134;"Points 40.
"
 2480  PRINTTAB(3,14);CHR$129;"FULL H
OUSE..two dice the saee and"'CHR$129;
"the other three the saee,i.e.2 three
s"'CHR$129;"and 3 fives.Points 40.";T
AB(3,17);CHR$130;"HI....Any fairly hi
gh score.Points"
```

```
 2490 PRINTCHR$130;"total of all dice
.MUST be higher than"'CHR$130;"LO."
 2500 PRINTTAB(3,20);CHR$131;"LO....A
ny low score.Points total of"'CHR$131
;"dice.MUST be lower than HI.";TAB(3,
22);CHR$133;"YAMS...all dice showing
sixes.Points"CHR$133;"60."
 2510  PRINTTAB(5,24);CHR$135;CHR$157
;CHR$132;"PRESS ANY KEY TO CONTINUE..
.."CHR$32;CHR$156;:AB$=GET$:CLS
 2520 PRINTTAB(6,11);CHR$141;CHR$134;
CHR$157;CHR$132;"You take your scores
 ";CHR$32;CHR$156;TAB(6,12);CHR$141;C
HR$134;CHR$157;CHR$132;"You take your
 scores ";CHR$32;CHR$156
 2530 PRINTTAB(1,14);CHR$141;CHR$134;
CHR$157;CHR$132;"froe the front face
of the dice.";CHR$32;CHR$156;TAB(1,15
);CHR$141;CHR$134;CHR$157;CHR$132;"fr
oe the front face of the dice.";CHR$3
2;CHR$156
 2540 PRINTTAB(3,24);CHR$131;CHR$157;
CHR$136;CHR$129;"PRESS ANY KEY TO CON
TINUE...."CHR$32;CHR$156;:AB$=GET$:CL
S:ENDPROC
 2550  DEFPROCerr
 2560  PRINTTAB(0,24);SPC(160);TAB(5,
24);"You've eisspelt soeething here o
r  left a space.... try again!"
 2570 SOUND0,-15,3,10:SOUND1,0,172,10
:ENDPROC
 2580 DATA ONES,TWOS,THREES,FOURS,FIV
ES,SIXES,TWO PAIRS,TRIPLE,RUN,FOUR KI
ND,FULL HOUSE,HI,LO,YAMS
```



George Cane puts an old favourite on your monitor

PATIENCE is a BBC Micro version of the long established card game which, by its nature, is for one player.

On screen you see seven vertical files of cards, all face down bar the last card of each file. The number of cards placed in each decreases by one as you look from left to right, resulting in there being only one card in file seven.

The remaining 24 cards form the pack. For convenience the pack is also laid in a vertical file on the extreme left of the screen and is numbered 1. The seven files of game cards are numbered 2 to 8.

Using these numbers you must move cards from one file to another in an attempt to finish with every card face up, and all the pack used up.

The all-important rule is that cards must be placed in the order "low" on "high" and in alternating colours. This means for example that a red 4 can only be placed on a black 5, a black queen can only be placed on a red king and so on.

When any vertical file becomes empty as a result of moving its last card, that empty space can be filled, but only with a king.

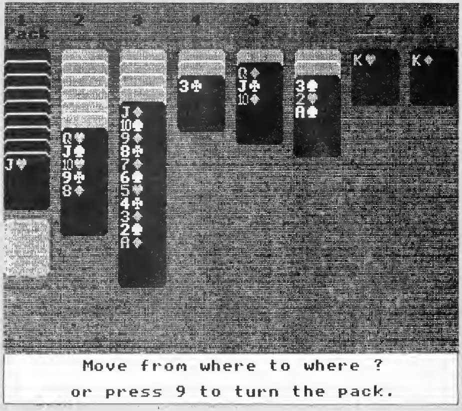
Just keep moving the cards until either you succeed in using all the pack and turning all 52 cards face up in ascending order, or you cannot move any card according to the rules.

By the way, the computer won't let you make an illegal move – though we're sure you wouldn't even try!

All you need is a little bit of luck . . . and an awful lot of patience.

```
LD 10 REM Patience
   20 REM (C) The Micro User
   30 *FXI1,0
   40 MODE1:VDU19,2,2,0,0,0:CDLOUR130
:CLS:PROCtext:VDU26:VDU23,1,0;0;0;0;:
PRDCintro:FORJ=0TO12:READA,B,C,D,E,F,
G,H:VDU23,224+J,A,B,C,D,E,F,G,H:NEXT:
DIMC%(52),F$(52),L%(7),L$(7),N$(7),V%
(52),V$(I3):FORA=1TD13:READV$(A):NEXT
:V$(10)=CHR$236
   50 T$=CHR$224+CHR$225+CHR$225+CHR$
226:M$=CHR$227+CHR$228+CHR$228+CHR$22
9:B$=CHR$230+"  "+CHR$231:FORA=0TO7:R
EADN$(A):NEXT
   60 VDU26:COLOUR130:PROCtext:PR1NT:
K$="Press any key to start":PRDCsay(3
):REPEATQ=RND:UNTILINKEY$(1)<>""
   70 K$="Do you Iike to play strictI
y ?":PRINT'':PROCsay(3):REPEATQ=RND:K
$=1NKEY$(1):UNTILK$<>"":J=0:1FK$="Y"J
=1
   80 PRINT'':K$="Would you Iike a di
fficult game ?":PRDCsay(3):REPEATQ=RN
D:K$=1NKEY$(1):UNT1LK$<>"":K=8:1FK$="
Y"K=9
   90 PROCshuffIe:PROCdeal:*FX21,0
  100 ONERRORGOTD590
  110 PROCplay:1FK$="A"ORK$="a"THEN90
:ELSEIFK$="C"ORK$="c"THEN70:ELSEGOTO1
00
  120 DEFPROCwindow:B%=W%*5:VDU28,B%,
28,B%+3,3:COLOUR130:CLS:ENDPROC
  130 DEFPROClong:VDU28,0,28,3,0:COLO
UR3:COLOUR130:CLS:PRINT'" 1  Pack";:E
NDPROC
  140 DEFPROCtext:VDU28,0,31,39,29:CO
LOUR128:CLS:ENDPRDC
  150 DEFPROCsay(c):COLOURc:PRINTTAB(
20-(LEN(K$)D1V2));K$;:ENDPRDC
  160 DEFPRDCshuffIe:VDU26:COLOUR130:
CLS:Q=RND(3):PROCtext:PRINT:IFH=0K$="
Remember, Press A for a new game.":PR
OCsay(3):PRINT':H=1
  170 1FQ<2K$="Just a quick shuffle."
:ELSEIFQ<3K$="I'm shuffling the pack.
":ELSEK$="l'm giving the cards a good
 shuffle."
  180 PROCsay(3):FDRA=0TO52:F$(A)="":
NEXT:FORA=1TO52:REPEAT:FORB=0TOQ:D%=R
ND(52):NEXT:UNTILF$(D%)="":C%=(A-1)DI
V13:V%(D%)=A MOD13:F$(D%)=V$(V%(D%)+1
)+CHR$(232+C%)+"  ":C%(D%)=C%AND1:NEX
T:ENDPROC
  190 DEFPROCpack:IFL%(0)=P%ENDPROC:E
LSECOLOUR129:CDLOUR2:PRINTT$"
    "B$;:ENDPROC
  200 DEFPROCone:COLOUR131:COLOUR2:PR
INTT$;:E%=VAL(LEFT$(L$(W%),2)):COLOUR
C%(E%):PRINTF$(E%)"          ";:COLOUR2
:PRINTB$;:ENDPROC
```

```
  210 DEFPROCtop:COLOUR131:COLOUR1:PR
INTT$;:COLOURC%(P%):PRINTF$(P%)"
   ";:COLOUR2:PRINTB$:ENDPROC
  220 DEFPROCIast:COLOURI31:COLOUR2:P
RINTM$;:COLOURC%(P%):PRINTF$(P%)"
    ";:COLOUR2:PRINTB$:ENDPROC
  230 DEFPROCdeal:VDU26:COLOUR130:COL
OUR3:CLS:PRINT:FORA=ITOB:PRINT" ";A;"
   ";:NEXT:PRINT"Pack":PROCtext:P%=52
:FORW%=1TO7:L$(W%)="":FORA=1TOK-W%:L$
(W%)=L$(W%)+STR$(P%):P%=P%-1:NEXT:L%(
W%)=A-1:PROCIay:NEXT
  240 L$="":L%(0)=0:FORA=ITOP%:IFA<I0
L$=L$+"0"
  250 L$=L$+STR$(A):NEXT:L$(0)=L$:PRO
Cturn:ENDPROC
  260 DEFPROCIay:PROCwindow:A%=LEN(L$
(W%))/2:IFA%<1ENDPROC:ELSEIFL%(W%)=A%
L%(W%)=L%(W%)-1:IFL%(W%)<0L%(W%)=0
  270 IFA%=IPROCone:ENDPROC
  280 IFL%(W%)<1THENCOLOURI3I:COLOUR2
:PRINTT$;:GOTO300:ELSECOLOUR129:COLOU
R2:PRINTT$;:IFL%(W%)=1THEN290:ELSECOL
OUR0:FORB=2TOL%(W%):PRINTM$;:NEXT
  290 COLOUR13I:COLOUR1:PRINTT$;
  300 COLOUR2:FORB%=L%(W%)*2+1TOA%*2:
E%=VAL(MID$(L$(W%),B%,2)):B%=B%+1:COL
OURC%(E%):PRINTF$(E%);:NEXT:COLOUR2:P
RINT"          ";B$;:ENDPROC
  310 DEFPROCkey:REPEAT:K$=INKEY$(100
):W%=VAL(K$)-1:UNTIL(W%>-IANDW%<9)ORK
$="A"ORK$="a"ORK$="C"ORK$="c":ENDPROC
  320 DEFPROCturn:W%=0:PROClong:IFP%=
0ENDPROC:ELSEL%(0)=L%(0)+3:IFL%(0)>P%
REPEAT:L%(0)=L%(0)-P%:UNTILL%(0)<=P%
  330 IFL%(0)=1PROCone:GOTO380:ELSEIF
L%(0)>I6CLS:IFL%(0)=17PRINT'" 1  ";:E
LSEIFL%(0)=18PRINT" 1  ";
  340 L%=20:IFL%(0)=P%L%=25
  350 IFL%(0)>1ANDL%(0)<L%COLOUR13I:C
OLOUR2:PRINTT$;
  360 IFL%(0)>1COLOUR131:COLOUR0:A=1:
REPEATA=A+1:PRINTM$;:UNTILA=L%ORA=L%(
0)
  370 IFL%(0)>1THENE%=VAL(MID$(L$(0),
L%(0)*2-1,2)):COLOURC%(E%):PRINTF$(E%
)"         ";:COLOUR2:PRINT8$;:COLOUR1
30
  380 PROCpack:ENDPROC
  390 DEFPROCfind:X%=VAL(RIGHT$(L$(W%
),2)):CN=C%(X%):N%=V%(X%):IFF%>0Y%=VA
L(MID$(L$(F%),L%(F%)*2+1,2)):CM=C%(Y%
):M%=V%(Y%):ENDPROC:ELSEY%=VAL(MID$(L
$(F%),L%(F%)*2-1,2)):CM=C%(Y%):M%=V%(
Y%):ENDPROC
  400 DEFPROCplay:PROCtext:K$="Move f
rom where to where ?":PROCsay(3):PRIN
T':K$="or press 9 to turn the pack.":
PROCsay(3):PROCkey:IFK$="A"ORK$="a"OR
K$="C"ORK$="c":ENDPROC:ELSEIFW%=8PROC
text:PROCturn:ENDPROC
```

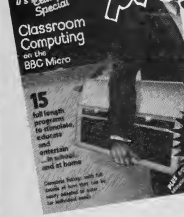


Move from where to where ?

or press 9 to turn the pack.

```
  410 IFW%=0ANDL%(W%)=0VDU7:K$="Turn
the pack first!":PROCtext:PRINT:PROCs
ay(2):GOTO500
  420 IFL$(W%)=""VDU7:K$="Thats silly
!.":PROCtext:PRINT:PROCsay(1):GOTO500
:ELSEF%=W%:COLOUR2:PRINTTAB(17,0)N$(F
%);:COLOUR3:PRINT" to where ?   ";:PR
OCkey:IFK$="A"ORK$="a"ORK$="C"ORK$="c
":ENDPROC:ELSEIFW%=8PROCtext:PROCturn
:ENDPROC
  430 IFW%=0K$="You may not put a car
d ON the pack!.":VDU7:PROCtext:PRINT:
:PROCsay(1):GOTO500
  440 PROCfind:IFL$(W%)=""THEN470:ELS
EIFM%=I2ANDL$(W%)<>""VDU7:PROCtext:PR
INT:K$="A King may only go to an empt
y space.":PROCsay(1):GOTO500
  450 IFN%<>M%+IVDU7:PROCtext:K$=V$(M
%+1)+"'s may only go on "+V$(M%+2)+"'
s.":PRINT:PROCsay(1):GOTO500
  460 IFCM=CN VDU7:PROCtext:PRINT:K$=
"Card colours may not be the same.":P
ROCsay(1):GOTO500
  470 IFL$(W%)=""ANDJ=1ANDM%<>I2VDU7:
PROCtext:PRINT:K$="Only a King may fi
II an empty space.":PROCsay(1):GOTO50
0
  480 K$="From "+N$(F%)+" to "+N$(W%)
+".":PROCtext:PRINT:PROCsay(2)
  490 PROCmove:ENDPROC
  500 FORDELAY=0TO5000:NEXT:ENDPROC
  510  DEFPROCmove:IFF%>0THEN550:ELSE
P$="":S$=MID$(L$(F%),L%(F%)*2-I,2):L%
(0)=L%(0)-1:P%=P%-1:IFP%=0L$(0)="":GO
TO560
  520 IFL%(0)>0P$=LEFT$(L$(0),L%(0)*2
):IFL%(0)=P%THEN540
  530 P$=P$+RIGHT$(L$(0),LEN(L$(0))-(
L%(0)+I)*2)
  540 L$(0)=P$:L%(0)=L%(0)-3:GOTO560
  550 S$=RIGHT$(L$(F%),LEN(L$(F%))-(L
%(F%)*2)):L$(F%)=LEFT$(L$(F%),L%(F%)*
2):L%(F%)=L%(F%)-I:IFL%(F%)<0L%(F%)=0
  560 L$(W%)=L$(W%)+S$:PROCIay:W%=F%:
IFW%=0PROCturn:ENDPROC:ELSEPROCIay:EN
DPROC
  570 DEFPROCintro:COLOUR130:PRINT'''
:K$="PATIENCE":PROCsay(3):PRINT'':K$=
"By G.T.Cane. Dec.'83.":PROCsay(3):PR
INT''''TAB(4);"This is a simulation o
f the well"''TAB(3);"known card game,
  The only rule is"''
  580 PRINTTAB(6)"'If you can do it,
 it's OK.'"''''':K$="If you get stuck,
 press A for another ":PROCsay(3):PRI
NT':K$="game or C to change the level
 of play.":PROCsay(3):ENDPROC
  590 IFERR=17VDU7:GOTOI00:ELSEPROCte
xt:REPORT:PRINT" on line ";ERL:END
  600 DATA&FF,&FF,&FF,&FF,&FF,&C0,&80
,0,&FF,&FF,&FF,&FF,&FF,0,0,0,&FF,&FF,
&FF,&FF,&FF,3,1,0,0,0,0,0,63,64,128,0
,0,0,0,0,255,0,0,0,0,0,0,252,2,1,0,
0,0,0,0,0,0,&80,&C0,0,0,0,0,0,0,I,3
  610 DATA28,28,107,127,107,8,28,0,8,
28,62,127,62,28,8,0,28,62,I27,127,127
,28,62,0,54,127,127,127,62,28,8,0,38,
I05,41,4I,4I,4I,118,0,A,2,3,4,5,6,7,8
,9,T,J,Q,K,One,Two,Three,Four,Five,Si
x,Seven,Eight
```

The best value ever in educational programs!


THE MICRO USER Education Special
Classroom Computing on the BBC Micro
15 full length programs to stimulate, educate and entertain ...in school and at home
Complete listings with full details of how they can be easily adapted to suit for individual needs

Available now at your newsagent

A must for every parent and teacher, the Micro User Educational Special will help turn your BBC Micro into a learning centre.

Its 15 original programs have been specially chosen to combine educational validity with sheer good fun.

Tested in the classroom, and ranging in scope from pre-reading to sixth form maths, each program is fully listed and explained in detail – in the clear style for which The Micro User is renowned.

And we've managed to squeeze all the programs from this Micro User Special into a software package that's overflowing with educational value.

Can you or your children afford to miss these vital educational resources?

Please send me the following:

□ I enclose cheque made payable to Database Publications Ltd: for £__________

□ I wish to pay by
□ Access □ Visa

No.__________

Expiry date __________

□ Education Special No.1	£1.95
□ BBC 'B' cassette	£5.95
□ BBC 40-track disc	£7.95
□ BBC 80-track disc	£7.95
Magazine PLUS software:	
□ Magazine and cassette	£7
□ Magazine and disc	£9

Please tick box

Signed __________

Name __________

Address __________

____

Post to: Database Publications, Europa House,
68 Chester Road, Hazel Grove, Stockport SK7 5NY.

SPEC. £5.95 CM64 £6.95
BINGO
UP TO 4 PLAYERS AT ONCE
A FIVE YEAR OLD OR GRANNY CAN PLAY
1
£100.00 TO BE WON EACH MONTH
FOR FURTHER DETAILS RING TREVOR ON 091 414 4611
66
ACTUALLY SHOUTS THE NUMBER OUT
CURRAH COMPATIBLE FOR CM64 & SPECTRUM
FUN FOR ALL THE FAMILY
TYNESOFT
COMPUTER SOFTWARE

**ADDISON INDUSTRIAL ESTATE, BLAYDON-ON-TYNE,
TYNE & WEAR. TEL: (091) 414 4611.**

ALL TITLES AVAILABLE FROM TYNESOFT
POSTAGE AND PACKING FREE

SUPERFILE 64
A REALLY USER FRIENDLY DATABASE FOR THE 64.
CM64 CASS **£17.95.** DISK **£19.95**

FIREMAN FRED
HELP FIREMAN FRED RESCUE CATS, DOGS & PEOPLE FROM 20 DIFFERENT LOCATIONS.
SPEC. **£5.95**

AUF WIEDERSEHEN
HELP OZ BUILD A WALL, DRINK HIS BEER, THEN FIND HIS WAY HOME.
SPEC. **£5.95** CM64 **£6.95**
BBC/ELECTRON **£6.95**

NEW! from SQUIRREL
Two very original programs for BBC-B and ELECTRON

POLAR PERILS £7.95
A 100% m/code multiscreen arcade/adventure game set in very hostile Arctic waters. No gun – just your wits and your spear for survival against many sub-zero hazards including ravenous polar bears who will hop on to your tiny ice floe without warning! Great skill is required to get through to the final screen of this very addictive adventure and only then do you realise that your difficulties have only just begun as you try to collect ice blocks in your kayak for your igloo home!

MONEYWISE £9.95
An up to date yet simple to use personal/home money management system based on not one but TWO main accounts with extensive analysis, budget forecasting, annual summary and 3D BAR CHARTS. Other facilities offered are – SAVE complete program to DISC, PASSWORD protected datafiles (tape or disc), PRINTOUTS of all accounts and 3D Bar chart screen dump.

Other established classics from SQUIRREL:-

TRAFALGAR £8.00
'A good combination of arcade action and strategy' – C & VG

SUPERGOLF £7.50
'I do know a good game when I see one and SUPERGOLF is just that!' – Electron User

BUNFUN £6.50
'Fast and furious – a game calling for quick reactions – fun to play' – The Micro User

All programs available on 40T disc – add £2.00

SQUIRRELSOFT

**Dept E,
4 BINDLOSS AVENUE, ECCLES, MANCHESTER M30 0DU
24 Hour answering service – 061-789 4120**

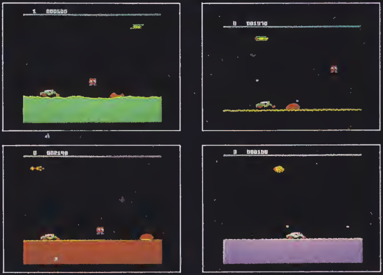
Cheques, P.O.s ▨ Same day despatch

SPECIAL OFFER! Deduct £1 per cassette when ordering 2 or more

THE BEST BBC MICRO SOFTWARE

* PRODUCED BY AN INDEPENDENT SOFTWARE HOUSE *
* TOP QUALITY MACHINE-CODE PROGRAMS *

BBC MICRO

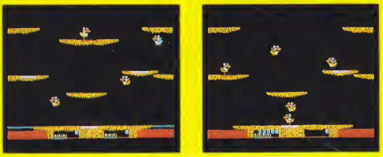


BMX ON THE MOON (32K) £7.95

Smooth scrolling-screen arcade action. In this game you control a moon cycle which you use to patrol the surface of the moon. You must avoid the rocks on the surface, and also the monsters bouncing high in the low gravity. You should try to shoot down the spaceships which fly overhead dropping bombs onto the moon surface: A further hazard are the flying barrels which are often just too high to shoot or jump over, and just too low to squeeze underneath.

●●●NEW RELEASE●●●

Many of our titles are available in Boots, W.H. Smiths, John Menzies, Rumbelaws, Laskys, Currys, HMV, Greens at Debenhams and Spectrum Shops. Also at all major computer dealers — Eltec Computers, Micro Management, West Coast Personal Computers, Microstyle, Electronequip, 3D Computers, Computerama, GTM Computers, etc. Our software is also available through all the major distributors, and directly from us by mail-order.



WINGED WARLORDS (32K) £7.95

A fascinating arcade-style game for one or two players. Each player rides on ostrich, and the aim is to knock the enemy riders off by colliding with them. In a collision the rider with the highest jousting pole will be victorious. If you are successful, the enemy will revert to its original form: an egg. The egg must be collided with to prevent it from hatching. Avoid the lava bath at the bottom of the screen, and remember: you can fly better by using stronger strokes — imagine you are an ostrich trying to fly!

(KEYBOARD OR JOYSTICKS)

●●●NEW RELEASE●●●

SUPERIOR SOFTWARE LTD.
Dept.TM11, Regent House,
Skinner Lane, Leeds 7
Tel: 0532 459453

OUR GUARANTEE

(1) All our software is available before we advertise.
(2) All our software is despatched within 48 hours by first-class post.
(3) In the unlikely event that any of our software fails to load, return your cassette to us and we will immediately send a replacement.

WIN! THIS FABULOUS GOLDEN TREASURE, THE FAMOUS GOLDEN JEWELLED HARE OF 'MASQUERADE' *OR* £30,000

WITH
HARERAISER
FROM
HARESOFT

Hareraiser, the unique computer treasure hunt, a computer puzzle of graphics and text in two parts each part contains vital clues to places or locations which will enable you to discover the whereabouts of this golden treasure. It has not been buried you just have to pin point its location to claim this amazing prize. See cassette inlay card rules for the treasure hunt of 1984. Part one (Prelude) available now. Part two (Finale) coming soon! Order yours now.

Available for

ELECTRON	BBC B	ORIC/ATMOS 48	DRAGON 32
SPECTRUM 48	CBM 64	VIC 20 EX	AMSTRAD
TANDY EXT COL 32K	ATARI 800 XL	MSX	

Available at £8.95 each (£17.90) total. From Harrods, W.H. Smith, Boots, Tandy, Currys Granada TV Rentals, Rumbelows, Lewis's Ltd, Debenhams, Lasky's and good Software outlets or direct from Haresoft Ltd. Access orders Phone 01-388 3910. 48 hr despatch.

HARESOFT LTD.
P.O. BOX 365, London NW1 7JD.
Tel: 01-387 5876